

Grado en Ingeniería Electrónica Industrial y Automática

Curso Académico 2011-2012

“DISEÑO MECÁNICO, IMPRESIÓN, Y MONTAJE DE UN VEHICULO IMPRIMIBLE”



**Universidad
Carlos III de Madrid**
www.uc3m.es

Trabajo Fin de Grado

Tutor: Alberto Valero Gómez

Autora: María Ramos Montero

Agradecimientos

En primer lugar, quiero agradecer al Departamento de Ingeniería de Sistemas y Automática junto con Alberto Valero y Juan González por la oportunidad que me han ofrecido para embarcarme en este proyecto y permitirme conocer el mundo de la impresión 3D. También cabe mencionar a mis compañeros Ana De Prado, Juan Carlos Rodríguez y Jose Manzano por compartir conmigo sus conocimientos durante todos estos años de carrera y darme apoyo en lo que necesitaba. A Alberto Calvo por acompañarme esas interminables horas de impresión, aconsejarme y ayudarme en el montaje del vehículo. A mi prima M^a Paz, por enseñarme con paciencia todos los entresijos de AutoCAD. Y, por último, a mis amigos y familiares, por estar ahí apoyándome y ayudando en todo lo que podían sin importar el tiempo dedicado o el esfuerzo necesario.

Resumen

Este proyecto es uno de los muchos que se han iniciado en la Universidad debido a la innovación que han provocado las impresoras *Thing-O-Matic*, del Departamento de Ingeniería de Sistemas y Automática, en el ámbito de la robótica. Además, tiene como objetivo probar la viabilidad de la robótica imprimible de bajo coste.

El objetivo es diseñar un vehículo utilizando como herramienta principal las impresoras 3D del departamento. Para ello, se accedió a un curso extraescolar gratuito impartido en la Universidad Carlos III de Madrid en el que enseñaban las herramientas, tanto de software como de hardware, que se necesitaban para hacer un correcto uso de las impresoras, diseñar robots y controlarlos.

Ello provocó un gran número de proyectos relacionados con los robots imprimibles. La novedad de este proyecto con respecto a los demás es el uso de motores de corriente continua, puesto que hasta ahora todos los robots imprimibles usaban servomotores para realizar tanto su movimiento como su desplazamiento. Además, la mayoría de ellos para realizar las acciones mencionadas y aprovechando el uso de los servomotores, han usado control diferencial mientras que en este proyecto se usa el control direccional, es decir, el usado por los coches comerciales.

Para el control del mismo se utilizará un gamepad, es decir, un mando de videoconsola, conectado al ordenador. Gracias a él, cualquier persona podrá manejar el robot de forma intuitiva incluso sin tener conocimiento alguno de robótica. Esto es debido al gran desarrollo que han tenido estos mandos con las nuevas generaciones de videoconsolas, lo que lo hacen un instrumento familiar y de fácil uso.

El vehículo utilizará comunicación inalámbrica mediante Bluetooth, permitiendo un movimiento más libre de hasta 9 metros de distancia al punto de control (ordenador) gracias a la supresión de los cables.

El elemento intermediario que recibirá las órdenes mandadas del gamepad mediante el Bluetooth es la placa Arduino Leonardo, hardware OpenSource al alcance de todos, que se encargará de procesar los datos recibidos y enviar las señales imprescindibles para que se mueva o reaccione el vehículo.

Abstract

This project is one of many that have been initiated at the University because of the innovation that has led printers Thing-O-Matic, by Department of Systems Engineering and Automation, in the field of robotics. Also it has the objective to test the viability of low cost printable robotics.

The objective is to design a vehicle using as main tool the department's 3D printers. To this end, agreed to a free-school course taught at the University Carlos III of Madrid where instructing you the tools, both software and hardware, which is needed to make proper use of printers, design and control robots.

This caused a large number of projects related to printable robots. The novelty of this project with respect to others is the use of motors DC, as so far all printable robots used servos for both, its movement and its displacement. Moreover, most of them for the actions mentioned and making use of the power units have used differential control, while in this is used directional control, the same that used by commercial vehicles.

To control it will use a gamepad, that is to say, a console controller, connected to the computer. With it, anyone can handle the robot intuitively even without any knowledge of robotics. This is due to the great developments that have these controls with the new generation of consoles, which make it a familiar instrument and easy to use.

The vehicle used by Bluetooth wireless communication, allowing freer movement of up to 9 meters away to the point of control (computer) through the removal of the cables.

The intermediary element receive gamepad orders sent by Bluetooth is the Arduino Leonardo, OpenSource hardware available to everyone, which will process the received data and send the necessary signals to move or react the vehicle.

Índice General

Agradecimientos	3
Resumen	4
Abstract	5
Índice General	6
Índice de Figuras	8
I. Memoria	10
1. INTRODUCCIÓN	11
1.1 Introducción	11
1.2 Objetivos del proyecto	11
1.3 Fases de desarrollo	11
2. IMPRESORAS 3D	13
2.1 Introducción	13
2.2 Definición	13
2.2.1 Breve Historia	14
2.2.2 Antecedentes en la Universidad Carlos III de Madrid	15
2.3 Características de las impresoras	16
2.4 Software utilizado	17
2.4.1 Replicatorg	18
2.4.2 OpenScad	20
2.4.3 OOML2	20
2.4.4 Qt Creator	21
3. TIPOS DE VEHÍCULOS	22
3.1 Introducción	22
3.2 Vehículos Acuáticos	22
3.3 Vehículos Aéreos	24
3.3.1 UAS	25
3.4 Vehículos Espaciales	26
3.5 Vehículos Terrestres	27
3.5.1 Vehículos terrestres sobre raíles	27
3.5.2 Vehículos Terrestres sin raíles	29
4. VEHÍCULO DESARROLLADO	32
4.1. Introducción	32
4.2. Elementos utilizados	32
4.2.1 Arduino Leonardo	32
4.2.2 Servomotores	35
4.2.3 Motores DC	36
4.2.4 Batería LIPO	37

4.2.5 Rodamientos	39
4.2.6 Circuito Integrado L293B	39
4.2.7 Control Remoto: GamePad	40
4.2.8 Módulo Bluetooth	41
4.2.9 Metacrilato	42
4.3. Diseño	43
4.3.1 Modelo seleccionado	43
4.3.2 Cinemática	44
4.3.3 Suspensión	48
4.3.4 Dirección	49
4.3.5 Reductor	51
4.3.6 Circuito de control de motores	55
4.4 Fabricación y Montaje	57
4.4.1 Montaje Parte Delantera	58
4.4.2 Montaje de la parte trasera	61
4.4.3 Montaje con la base	63
4.5 Implementación	64
4.5.1 Control mediante GamePad	65
4.5.2 Comunicación inalámbrica	67
4.5.3 Pruebas y modificaciones	69
5. CONCLUSIONES Y TRABAJOS FUTUROS	71
6. BIBLIOGRAFÍA	73
II. Planos	75
1. INTRODUCCIÓN	76
2. PARTE DELANTERA	77
3. PARTE TRASERA	88
III. Anexos	100
A. HOJAS DE CARACTERÍSTICAS	101
A.1 Motor rs_540sh	101
A.2 Diodo 1N4001	102
A.3 Circuito Integrado L293B	104
A.4 Atmega32u4	113
B. PLANIFICACIÓN DE TAREAS Y PRESUPUESTO	118
B.1 Organización de Tareas	118
B.2 Presupuesto	123

Índice de Figuras

Figura 1. Impresión por fotopolimerización	14
Figura 2. Impresora 3D Contex.....	15
Figura 3. Robots imprimibles o PrintBots desarrollados en la UC3M.....	15
Figura 4. Impresora Thing-o-Matic MADRE	16
Figura 5. Impresora Thing-o-Matic UC3-PO.....	17
Figura 6. Botón de conexión con la impresora	18
Figura 7. Entorno Replicatorg. Selección de botones para modificar diseño	18
Figura 8. Botón para acceder al menú de temperaturas y posición de la impresora.....	19
Figura 9. Menú para indicar la temperatura de la impresora	19
Figura 10. Pieza creada con OpenScad	20
Figura 11. Tipos de vehículos acuáticos.....	22
Figura 12. Tipos de vehículos sumergibles	23
Figura 13. Coches anfíbio.....	24
Figura 14. Tipos de fuselaje.....	24
Figura 15. Interior de un vehículo no tripulado	25
Figura 16. Vehículos no tripulados	26
Figura 17. Simulación del aterrizaje de Curiosity en Marte.....	27
Figura 18. Diferentes tipos de trenes.....	28
Figura 19. Tren no tripulado.....	29
Figura 20. Suspensión MacPherson	30
Figura 21. Automóvil todoterreno y buggy	31
Figura 22. Stanley, ganador Grand Challenge 2005	31
Figura 23. Arduino Leonardo.....	32
Figura 24. Entorno Arduino 1.0.1	34
Figura 25. Partes de un servo.....	35
Figura 26. Señal y posición del servo	35
Figura 27. Servo Futaba s3003	36
Figura 28. Funcionamiento de un motor DC.....	37
Figura 29. Batería LIPO Zippy	38
Figura 30. Rodamientos rígidos de bola.....	39
Figura 31. Esquema Puente H	39
Figura 32. Circuito integrado L293B	40
Figura 33. Mando Xbox 360	41
Figura 34. Módulo Bluetooth JY-MCU.....	42
Figura 35. Tipos de robots móviles	43
Figura 36. Restricciones cinemáticas	45
Figura 37. Sistemas de coordenadas	45
Figura 38. Análisis cinemático.....	46
Figura 39. Comparación de suspensión	48
Figura 40. Suspensión realizada por Daniel Gómez	48
Figura 41. Parametric airless tire	49
Figura 42. Modelo de dirección Ackerman.....	50
Figura 43. Sistema de dirección con alambre.....	51
Figura 44. Partes de un engranaje.....	52
Figura 45. Sistema Reductor obtenido.....	54
Figura 46. Esquema del circuito	55

Figura 47. Resultado del circuito	56
Figura 48. Montaje Suspensión I parte	58
Figura 49. Montaje Suspensión II parte	58
Figura 50. Montaje piezas delanteras	59
Figura 51. Montaje pieza dirección con rodamiento y rueda	59
Figura 52. Montaje pieza lateral con pieza dirección	60
Figura 53. Montaje del servo	60
Figura 54. Unión de la parte delantera con el soporte del servo	61
Figura 55. Resultado final de la parte delantera	61
Figura 56. Colocación engranajes en soporte trasero derecho	62
Figura 57. Colocación de engranajes y motor	62
Figura 58. Montaje de engranajes y rueda trasera	63
Figura 59. Configuración de la placa de metacrilato	63
Figura 60. Montaje completo del coche	64
Figura 61. Diagrama de clases	65
Figura 62. Flujograma del programa de C++	66
Figura 63. Botones usados del gamepad	67
Figura 64. Diagrama de bloques del funcionamiento del sistema	68
Figura 65. Conexión total del sistema	68
Figura 66. Algunas de las modificaciones realizadas	70
Figura 67. Brazo Robótico y Vehículo	70

I. Memoria

1. INTRODUCCIÓN

1.1 Introducción

En este primer capítulo de la memoria se va a proceder a describir los objetivos del Trabajo Fin de Grado y las fases de desarrollo que se han seguido para conseguir la implementación del mismo.

1.2 Objetivos del proyecto

- Diseño mecánico de un vehículo imprimible.
- Capacidad del vehículo de transportar un brazo robótico a escala.
- Utilización de dos motores de continua con control direccional.
- Control remoto del vehículo mediante un mando de videoconsola.
- Comunicación inalámbrica mediante un módulo de Bluetooth.

1.3 Fases de desarrollo

Las fases que se han seguido para realizar este Trabajo Fin de Grado han sido las siguientes:

- *Diseño mecánico del vehículo:* En esta fase se hizo un estudio del software y hardware necesario para realizar diseños imprimibles. De esta forma podíamos ver las limitaciones que planteaban y empezar a esbozar el vehículo.
- *Diseño de la electrónica:* Para poder controlar los motores de corriente continua hizo falta implementar un circuito impreso o PCB (del inglés: *Printed Circuit Board*) que contuviera el circuito integrado L293B.
- *Programación del control remoto:* Para la comunicación del mando de control con el ordenador en lenguaje C++.
- *Programación del control del vehículo:* Ante los datos recibidos por el mando, programar en Arduino las diferentes acciones que se necesitan que haga el coche para cumplir los objetivos marcados.

- *Comunicación inalámbrica:* Conseguir que el ordenador y Arduino se comuniquen entre sí mediante el módulo de Bluetooth para evitar las limitaciones de distancia que producen los cables.
- *Pruebas:* Confirmar el correcto funcionamiento y corregir los errores que van surgiendo.
- *Redacción y desarrollo de la memoria*

2. IMPRESORAS 3D

2.1 Introducción

En este capítulo se va a describir lo que es una impresora 3D, su funcionamiento y software necesario para su funcionamiento y para el diseño de piezas en 3D. Este tipo de impresora ha sido una parte fundamental en este proyecto y, por ello, se le dedica todo un capítulo.

2.2 Definición

Una impresora 3D es una máquina que es capaz de “imprimir” diseños en 3D, creando piezas tridimensionales a partir de un diseño hecho por ordenador. De esta forma se consiguen obtener prototipos reales.

Desde su creación, las impresoras han experimentado una gran evolución, habiéndose desarrollado varios tipos:

- ***Impresión por deposición de material fundente***

Es el método usado por las impresoras *Thing-O-Matic* del Departamento de Ingeniería de Sistemas y Automática y la mayoría de impresoras 3D de bajo coste. Consiste en que la impresora crea un modelo de la pieza a imprimir por capas, por lo que posteriormente la impresora va rellenando de plástico la pieza capa a capa.

Otro enfoque es fundir de manera selectiva el medio de impresión sobre una base. En esta variación el medio no fundido sirve de soporte para los resaltes y paredes delgadas de la pieza a producir, reduciendo así la necesidad de soportes auxiliares temporales. Típicamente un láser es usado para sinterizar el medio y formar el sólido.

- ***Impresión por procesamiento digital de luz***

Este procedimiento se basa en colocar un recipiente de polímero líquido expuesto a la luz de un proyector de procesamiento digital de luz en condiciones controladas, de forma que el polímero expuesto termina endureciendo. Así, la placa que contiene el recipiente, se mueve en pequeños incrementos hasta que el modelo se endurece por completo y el polímero restante queda en estado líquido.

- ***Impresión por inyección***

Es una técnica en la que se van esparciendo capas de polvo en la base y se inyecta un unificador en la sección de la pieza, por lo que al retirar todo el polvo de alrededor queda la

pieza como parte sólida. Gracias a esta tecnología se consiguen imprimir salientes o extraplanos.

- **Fotopolimerización**

Suele usarse cuando se tiene mucho detalle. La fotopolimerización por absorción de fotones consiste en un bloque de gel en el que un láser va trazando la pieza, solidificando las partes que enfoca. Este método ofrece tamaños de menos de 100 nm.

2.2.1 Breve Historia

El concepto de la impresión 3D por capas comenzó en 1986 por la empresa 3D Systems. Esta empresa usaba una técnica llamada “estereolitografía” cuyo autor, Charles Hull, lo patentó como la fabricación de objetos sólidos mediante la sucesiva “impresión” de delgadas capas una sobre otra. La estereolitografía usaba la impresión 3D mediante fotopolimerización (figura 1).

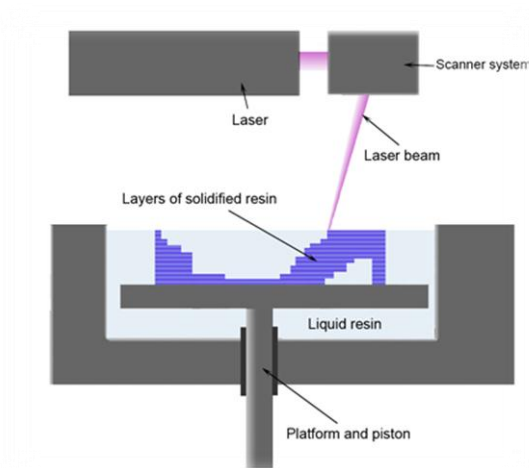


Figura 1. Impresión por fotopolimerización

Esta idea solo era concebible para las grandes empresas, puesto que los diferentes tipos de impresoras que se desarrollaban eran de muy alto coste. Gracias a la MIT (Massachusetts Institute of Technology) eso cambió. Dos de sus investigadores fundó Z Corporation, cuya empresa comenzó a desarrollar la tecnología que ahora usan las impresoras 3D comerciales. Posteriormente esta empresa fue comprada por Contex la cual dispone de una línea de distribución internacional. En la figura 2 podemos apreciar una de las impresoras que desarrollaron usando la impresión por inyección:



Figura 2. Impresora 3D Contex

A partir de ahí se comenzaron a desarrollar las impresoras de bajo coste, que en estos últimos años han disminuido mucho su precio. Gracias a ello, se dispone de dos impresoras en el Departamento de Ingeniería de Sistemas y Automática que nos permiten continuar investigando y desarrollando nuevas ideas y conceptos.

2.2.2 Antecedentes en la Universidad Carlos III de Madrid

Desde el curso 2010/2011 disponemos en la universidad de impresoras 3D, gracias a las cuales se han conseguido desarrollar una flota de robots variados como los que se muestran en la figura 3:

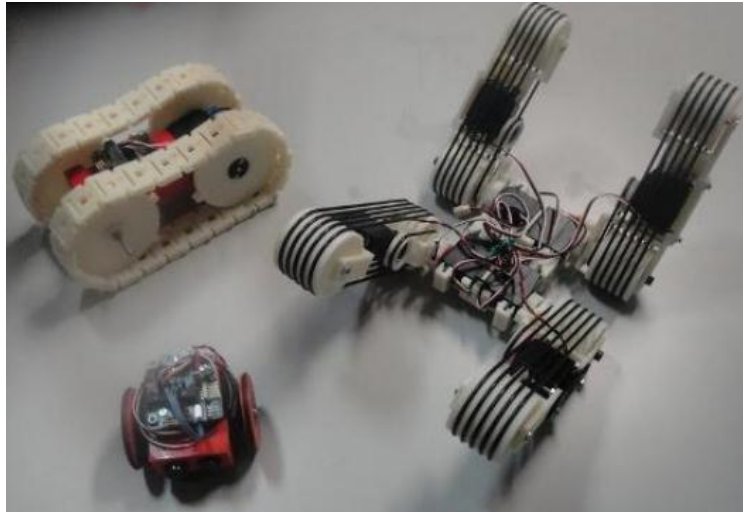


Figura 3. Robots imprimibles o PrintBots desarrollados en la UC3M

Los robots mostrados en la figura 3 son la primera generación de robots creados en la Universidad Carlos III de Madrid. Podemos apreciar de izquierda a derecha al Orugator (de Olalla Bravo y Daniel Gómez), Miniskybot y el F-Track (de Jon Goitia). A partir de ellos, han ido enseñando a aquellos que querían saber más sobre ese mundo, fuera del ámbito que fueras, compartiendo así sus conocimientos con el resto de la gente. De esta forma, hemos creado una cadena para que se expanda más rápidamente y a un mayor número de personas todo lo relacionado con el estudio en 3D.

Con ello se ha conseguido tener otra perspectiva de la robótica, produciendo un especial auge e interés en esta materia y aumentando así el número de seguidores de estas tecnologías. Además, dicha tecnología tiene la característica de ser OpenSource, es decir, hardware y software libre para poder copiarlo, modificarlo o venderlo. Por lo que todo el mundo puede acceder a ella, compartirla y mejorarla en cualquier parte del mundo. Se ha creado una gran comunidad en internet, con la cual, la innovación, creatividad y desarrollo no tiene fin.

Gracias a las impresoras, hemos pasado de tener robots solo disponibles en las simulaciones de ordenador, a poderlos crear y tener físicamente. La UC3M ha sido pionera en este aspecto y, gracias al éxito conseguido con estos proyectos, se ha expandido a otras universidades como, por ejemplo, la Universidad Politécnica de Madrid.

2.3 Características de las impresoras

Las impresoras disponibles en la Universidad por el Departamento de Ingeniería de Sistemas y Automática tienen las siguientes características:

- **Nombre:** MADRE
- **Familia:** RepStrap
- **Generación:** 0
- **Modelo:** Thing-o-matic, de Makerbot Industries (Número 4224)
- **Electrónica:** Arduino mega 2560 + Electrónica Gen 4
- **Extrusor:** MK6, con aguja de 0.5mm, Plástico de 3mm
- **Tamaño de la base:** 12x12 cm
- **Nacimiento:** 16/Mayo/2011 (Impresión de la primera pieza)



Figura 4. Impresora Thing-o-Matic MADRE

- **Nombre:** PADRE (a.k.a UC3-PO)
- **Familia:** RepStrap
- **Generación:** 0
- **Modelo:** Thing-o-matic, de Makerbot Industries
- **Electrónica:** Arduino mega 2560 + Electrónica Gen 4
- **Extrusor:** MK7, Aguja de 0.4, Plástico de 1.75mm
- **Tamaño de la base:** 12x12 cm
- **Nacimiento:** 8/Octubre/2011



Figura 5. Impresora Thing-o-Matic UC3-PO

La impresora MADRE fue la primera que se compró y su nombre se debe a que ha sido la creadora de otras muchas impresoras del proyecto Clone Wars. UC3-PO se adquirió unos meses más tarde, pero la mejora que habían hecho en el modelo era importante. Además de necesitar un plástico de diámetro más fino y tener otro modelo de extrusor, la precisión y velocidad eran mucho mayores que MADRE. De esta forma conseguíamos un acabado en las piezas mucho mejor. También cabe mencionar, que su capacidad para alcanzar los valores de temperatura necesarios para comenzar a imprimir, es mucho más rápida, necesitando así un menor tiempo de preparación. Por lo que conseguimos una mayor eficiencia de tiempo de impresión que con la primera.

2.4 Software utilizado

A continuación se detalla el software utilizado para poder llevar a cabo este proyecto y necesario para la impresión en 3D:

2.4.1 Replicatorg

Replicatorg es el programa encargado de leer los archivos .stl (tipo de archivo donde se guardan los diseños en 3D) y crear los .gcode (documento donde se indica las características de la impresión como velocidad, relleno, capas, etc.). Además, permite modificar el diseño creado en 3D escalándolo, rotándolo o moviéndolo para una mejor impresión.

Con este programa conseguimos controlar nuestra máquina, es decir, controlar la temperatura deseada, la posición a la que se establece el extrusor y cargar el nuevo firmware para las tarjetas de Arduino integradas en la impresora.

Para realizar una impresión con este programa se deben de seguir los siguientes pasos:

- Abrimos el programa
- Pulsamos el botón de conexión de la impresora (figura 6), para que el programa establezca comunicación con ella. Si todo va correctamente, la barra que aparece de color amarillo en la imagen, se pondrá de color verde.



Figura 6. Botón de conexión con la impresora

- Abrimos nuestro archivo .stl pulsando en el Menu/File/Open.../La carpeta donde se encuentre el fichero. Una vez seleccionado deberá aparecer el diseño en 3D en la pantalla.
- Modificamos nuestro diseño, si fuera necesario, ayudándonos de los botones que aparecen en la parte abajo a la derecha, como se indica en la figura 7:

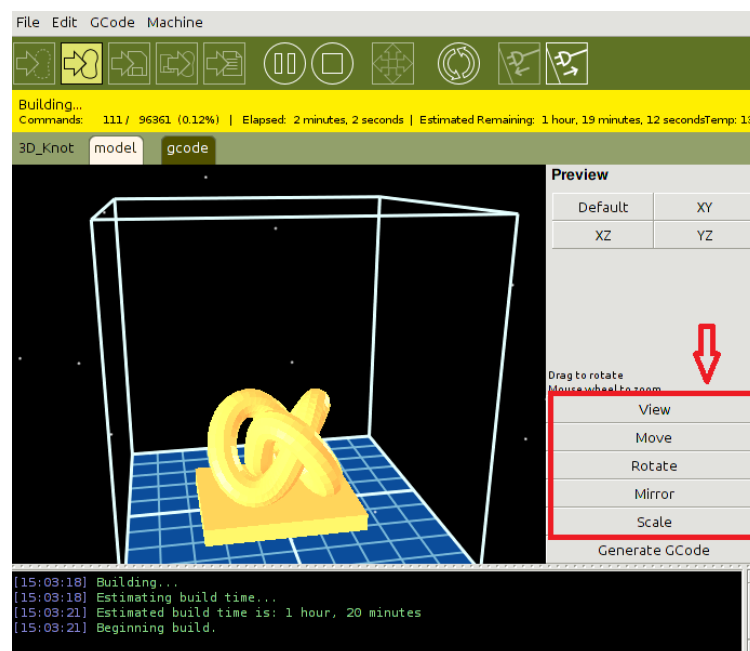


Figura 7. Entorno Replicatorg. Selección de botones para modificar diseño

- El siguiente paso es calentar la plataforma y el extrusor a las temperaturas adecuadas. Para ello, seleccionamos el botón identificado como una cruz de flechas (figura 8).



Figura 8. Botón para acceder al menú de temperaturas y posición de la impresora.

Una vez entramos, nos aparecerá un menú parecido al mostrado en la figura 9 (depende de la versión que se utilice). Debemos poner la temperatura del extrusor en el espacio marcado como “Target Temperature”. Lo aconsejable es una temperatura entre los valores de 220 a 240°C, según sea el cabezal. También deberemos escribir la temperatura para la plataforma de la impresora en el espacio de “Platform Target Temp”. En este caso, el valor aconsejado es de 120°C. Abajo nos aparecerá una gráfica en la que nos indica cómo se va calentando cada componente.

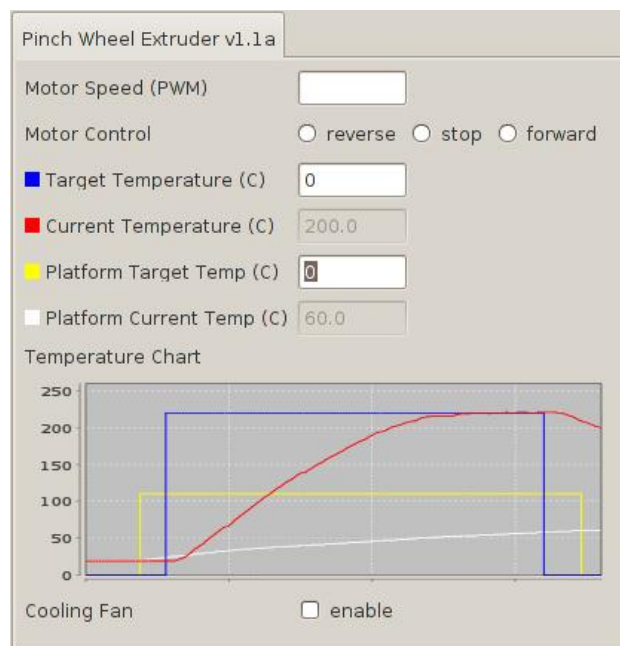


Figura 9. Menú para indicar la temperatura de la impresora

- Una vez dejamos calentando los componentes y tenemos el diseño como queremos, creamos el GCode de la figura. Para ello pulsamos el botón “Generate GCode” en la parte inferior derecha. Nos aparecerá un menú donde podremos elegir la velocidad de impresión, si queremos raft (una primera capa gruesa para evitar que se levante la pieza durante la impresión), el porcentaje de relleno que queremos que tenga la pieza, etc. Una vez rellenamos todo damos a crear. Según la complejidad de la figura, tardará entre unos segundos o algunos minutos en hacerse.
- Por último, pulsamos el botón de “Build”, el primero de la serie de iconos anteriormente mostrada (figura 7) y comenzará a imprimir.

2.4.2 OpenScad

OpenScad es un programa para la creación de objetos 3D CAD. Es software libre, por lo que está disponible en todas las plataformas y accesible para todo el mundo. OpenScad no se centra en los aspectos artísticos de modelado, sino en la funcionalidad y en crear piezas con finalidad más técnica, como piezas de maquinaria.

La creación de piezas es mediante programación. En varios comandos describes la pieza que deseas y el programa crea el modelo 3D a partir del archivo de comandos. De esta forma, el diseñador tiene el control total sobre la pieza, pudiendo modificar cualquier paso cuando desee o crea necesario. Se dispone de los sólidos primitivos (cubos, esferas, cilindros, polígonos, etc.) a partir de los cuales mediante funciones matemáticas, transformaciones o proyecciones podemos conseguir piezas tan complejas como deseemos. En la figura 10 podemos apreciar una rueda para servo creada con este programa:

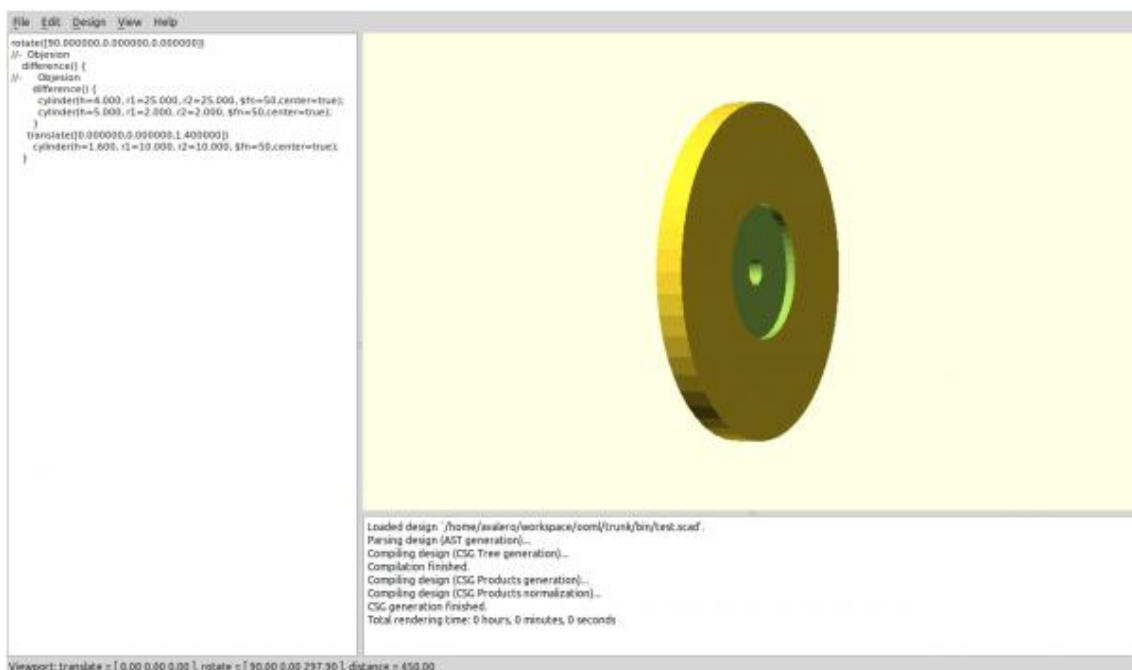


Figura 10. Pieza creada con OpenScad

Además, con él conseguimos el archivo .stl necesario para que lo lea el programa Replicatorg. Para ello debemos ir al menú que aparece en la parte superior del programa, Design/Export as STL...

2.4.3 OOML2

A diferencia de OpenScad, OOML2 es una librería de lenguaje de programación de C++, como su propio nombre indica: The C++ Object Oriented Mechanics Library. Ha sido desarrollada en la Universidad Carlos III de Madrid y sus creadores son Juan González Gómez , Alberto Valero Gómez , Rafa Treviño y Mario Almagro.

Las ventajas de usar este lenguaje para el diseño de piezas son las siguientes:

- Se *simplifica* en gran medida el lenguaje de comandos de OpenScad, consiguiendo un código de rápida compilación. Además, OOML2 permite tener un mayor orden del código y dar nombre a las diferentes partes de las piezas. Gracias a ello, es más fácil guiarse ante figuras complejas y que otras personas lo entiendan mejor.
- Es *OpenSource*, de forma que todo el mundo puede acceder a esta librería teniendo cualquier compilador de C++. Así, la gente puede ampliar dicha librería con los objetos que crea oportunos y compartirlo.
- Posee *objetos complejos* guardados además de los primitivos, como un soporte para pilas, servos futaba 3003, ruedas para servos, etc.

Las desventajas que presenta es que el programa sólo crea el archivo .scad, por lo que para conseguir el documento con expansión .stl para imprimir debemos abrir el programa OpenScad y crearlo desde ahí. Además, no tienes ninguna referencia visible mientras programas de cómo está quedando la pieza hasta que abres OpenScad.

En este proyecto, la mayoría de piezas han sido diseñadas mediante programación OOML2 debido a la sencillez del código incluso con piezas complejas. Las otras fueron hechas directamente en OpenScad debido a que se estaba terminando de desarrollar OOML2 y no estuvo disponible hasta un tiempo más tarde.

2.4.4 Qt Creator

Qt Creator es un entorno de desarrollo integrado (IDE) que le proporciona las herramientas necesarias para diseñar y desarrollar aplicaciones. Este programa ha sido usado como interfaz para programar en los lenguajes C, C++ y OOML2. El motivo por el que se escogió fue porque puede ser usado en cualquier plataforma, y proporciona sencillez y facilidad al programar gracias a que las características del editor incluyen resaltado de sintaxis y autocompletado.

Además, incluye un gestor de proyectos donde se recoge la información de qué archivos lo componen, los ajustes seleccionados, etc. Ello hace que se vea de forma clara, dentro del programa, las clases o archivos de las que está formada nuestro programa y poder modificarlas si es necesario.

Es capaz de crear aplicaciones para cualquiera de los entornos (Windows, Linux, Mac) e incluso para móviles (Symbian, Maemo y MeeGo).

3. TIPOS DE VEHÍCULOS

3.1 Introducción

En este capítulo se hará un estudio de los diferentes tipos de vehículos que existen y las características generales de cada uno. Se hará especial mención a los vehículos terrestres puesto que es el que se ha seleccionado para este proyecto.

Antes de comenzar con los tipos de vehículo, se debe aclarar que es un vehículo: Según la RAE es un “Medio de transporte de personas o cosas.” Por lo que con esta definición podemos decir que un vehículo es un aparato o máquina de cualquier tipo o forma que se utiliza para transportar cargas o personas, con o sin motor.

3.2 Vehículos Acuáticos

Los vehículos acuáticos cubren una gran variedad de tipos como son los barcos, aerodeslizadores o los submarinos. Se usan para las actividades marítimas de educación o el ocio, la pesca y la extracción de recursos, el transporte de mercancías o de pasajeros, y para llevar a cabo el combate o las operaciones de salvamento. Es decir, tenemos una embarcación distinta para la actividad que necesitemos desempeñar.



Figura 11. Tipos de vehículos acuáticos

Para diseñar un vehículo de este tipo, las características fundamentales a tener en cuenta son la capacidad de carga, la velocidad y la navegabilidad. Depende de la función que vaya a desempeñar, predominará una de las características mencionadas sobre las demás. Por ejemplo, para un buque de guerra, la velocidad es lo más importante. Para embarcaciones de ocio, la seguridad es la que predomina. Eso sí, todos deberán cumplir sus respectivas normativas, como la de naufragios para grandes embarcaciones. Ahora existen programas de ordenador, en los que según diseñas los vehículos puedes hacer simulaciones para comprobar que cumple todo lo requerido.

Uno de los ámbitos donde más se ha desarrollado la tecnología ha sido en la propulsión. Antiguamente, los barcos eran propulsados por remos, palas o vela. Fue evolucionando con la máquina de vapor y, posteriormente, los motores de combustión interna, los cuales son los que predominan. Actualmente, como en cualquier ámbito, se buscan fuentes baratas, renovables y menos contaminantes que las que hay.

Otro de los ámbitos donde más ha evolucionado la tecnología ha sido en la navegación. De los antiguos astrolabios para poder mirar la posición de las estrellas y a partir de la intersección de varias de ellas poder averiguar nuestra posición (observación astronómica), a los instrumentos analógicos y digitales que dependen del sistema GPS, y nos muestran con exactitud donde nos encontramos en cada momento.

Hemos de diferenciar dos grandes grupos de estos vehículos: Los que se encuentran sobre la superficie del agua y los que se encuentran bajo ella. Todos los aspectos mencionados anteriormente son muy diferentes entre estos dos grupos. Por ejemplo, los vehículos que se mueven bajo el agua han de ser diseñados para soportar altas presiones, tener capacidad para dar oxígeno y tener gran navegabilidad. Y según si son teleoperados o tripulados, deberán llevar unos equipos a bordo acorde con la función a desempeñar. Por ejemplo, en la figura 12 podemos apreciar los diferentes vehículos marinos que se han creado para ir bajo la superficie y que profundidad alcanzan:

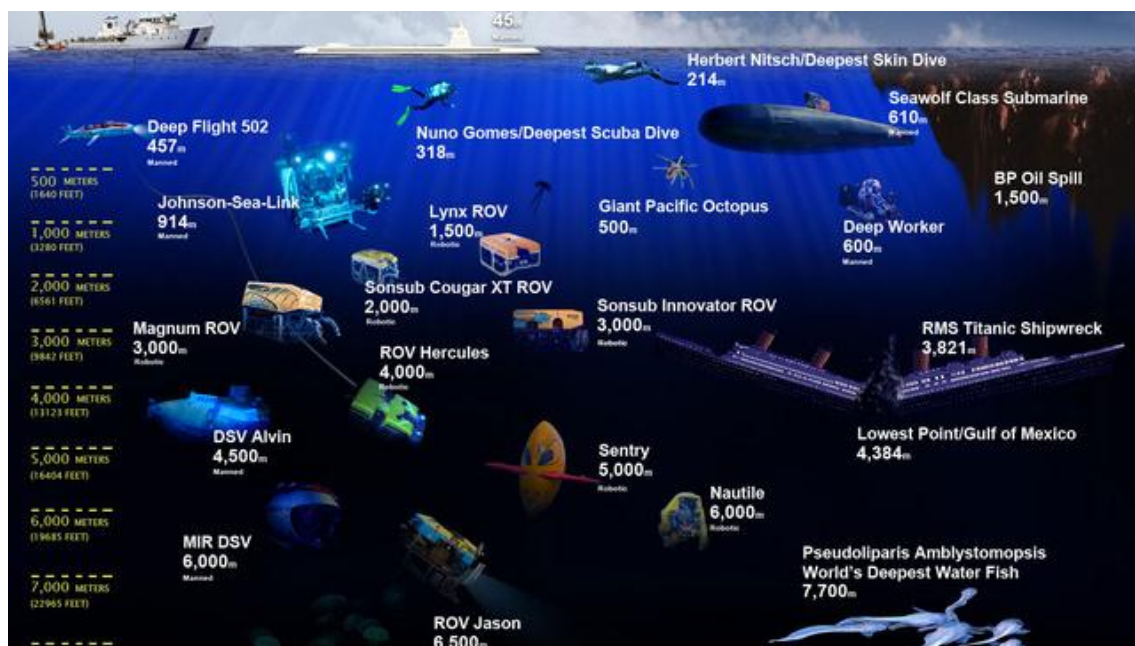


Figura 12. Tipos de vehículos sumergibles

Por último, comentar que en estos últimos años se están desarrollando los híbridos de tierra y agua (también llamados *anfíbios*), es decir, coches que cuando entran en el agua se transforman para que puedas manejarlos como una lancha o submarino. Por ejemplo, tenemos el curioso caso del Squba (figura 13), inspirado en una de las películas de James Bond, puede transformarse al presionar un botón en un auto anfibio que puede sumergirse hasta

100 metros bajo el agua. O el Aquada, un coche que se convierte en una lancha una vez te metes en el agua.



Figura 13. Coches anfibio

3.3 Vehículos Aéreos

Los vehículos aéreos son los aviones, también denominados aeroplanos o aeronaves con mayor densidad que el aire, provistos de alas y un espacio de carga capaz de volar, impulsados por uno o más motores. Su principio de funcionamiento se basa en la fuerza aerodinámica que se genera sobre las alas en sentido ascendente (sustentación). Esto es originado por la diferencia de presiones entre la parte superior e inferior del ala.

Al igual que los vehículos acuáticos, los aéreos tienen un gran número de funciones y según cada una tienen un diseño y estructura diferentes. Donde más se aprecia es en el fuselaje o cuerpo del avión al que se encuentran unidos las alas y los estabilizadores tanto horizontales como verticales. Su interior es hueco, para poder albergar dentro la cabina de pasajeros, la de mandos y los compartimentos de carga. En la figura 14 podemos apreciar algunos de los diferentes fuselajes, donde el número uno es para vuelo subsónico, el dos para vuelo supersónico (capaz de sobrepasar la barrera del sonido) de alta velocidad, el tres para vuelo subsónico con capacidad para grandes cargas, el cuarto para vuelos supersónicos de gran maniobrabilidad, el quinto para hidroaviones y el sexto para vuelo hipersónico (velocidades superiores a cinco veces la velocidad del sonido).

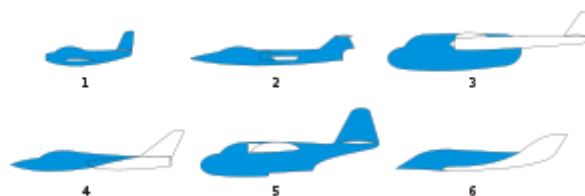


Figura 14. Tipos de fuselaje

Pero sin duda, independientemente del tipo de avión, la parte más importante son las alas, puesto que son las que sustentan la aeronave en el aire gracias a la curvatura de la parte superior de las mismas que hace que el aire por encima de ellas se acelere. Mientras que en la parte inferior suele ser plana, haciendo que el aire mantenga la misma velocidad y presión del aire relativo consiguiendo una mayor sustentación, creando una fuerza que contrarresta la

gravidad. Además, se suele variar la configuración del ala para ciertas acciones mediante los alerones, flaps, spoilers o slats. Cada uno realiza una función diferente: Los alerones están situados en los extremos del ala y gracias a ellos el avión puede girar en el eje horizontal (también llamado alabeo). Los flaps están situados al borde de salida del ala y son utilizados sobre todo para el despegue y el aterrizaje puesto que son hipersustentadores (aumentan el sustentado). Los spoilers están situados en la parte superior del ala y se usan para reducir la sustentación, puesto que cuando son extendidos hacen entrar en pérdida de forma que aumentan la velocidad de descenso. Y por último los slats, que también son hipersustentadores, pero están situados en la parte delantera del ala, de forma que cuando se usan aumentan la curvatura del ala, impidiendo la caída del avión incluso con velocidades reducidas.

Otros componentes importantes son los motopropulsores, ya que son los encargados de generar la tracción necesaria para contrarrestar la resistencia aerodinámica. Los motores son probados por los fabricantes para evitar problemas durante el vuelo. Como dato curioso, las últimas pruebas que tenía que pasar un motor es que un pollo (ya muerto) lo atravesara sin que se rompiera, debido al gran número de accidentes causados por aves que eran absorbidas por el motor.

Por último, mencionar los estabilizadores tanto horizontales como verticales, situados normalmente en la parte trasera del fuselaje. Su función es permitir los movimientos de cabeceo (girar hacia arriba o hacia abajo) y “guiñada” (hacia izquierda o derecha).

3.3.1 UAS

Dentro de los vehículos aéreos, durante las últimas décadas ha tenido un especial auge el estudio y desarrollo de los UAS (Unmanned Aerial System) o vehículos aéreos no tripulados. También son llamados UAV (Unmanned Aerial Vehicle), pero debido a la complejidad de los componentes y la cantidad de equipos que llevaban a bordo, se cree más correcta la primera nomenclatura.

Han generado un especial interés puesto que gracias a ellos se puede realizar misiones peligrosas o de espionaje sin miedo a que las personas que tripulan el avión salgan heridas o sean raptadas por el enemigo.



Figura 15. Interior de un vehículo no tripulado

Estos equipos de una tecnología muy avanzada, puesto adquieren altas velocidades y requieren de un control muy sofisticado y cuidado para evitar accidentes. Parte de los avances que se hacen en este campo se incorporan a los aviones tripulados para aumentar la seguridad de los aparatos.

En la figura 16 se pueden ver los diferentes de vehículos no tripulados que se han hecho, y la altura a la que operan y su durabilidad:

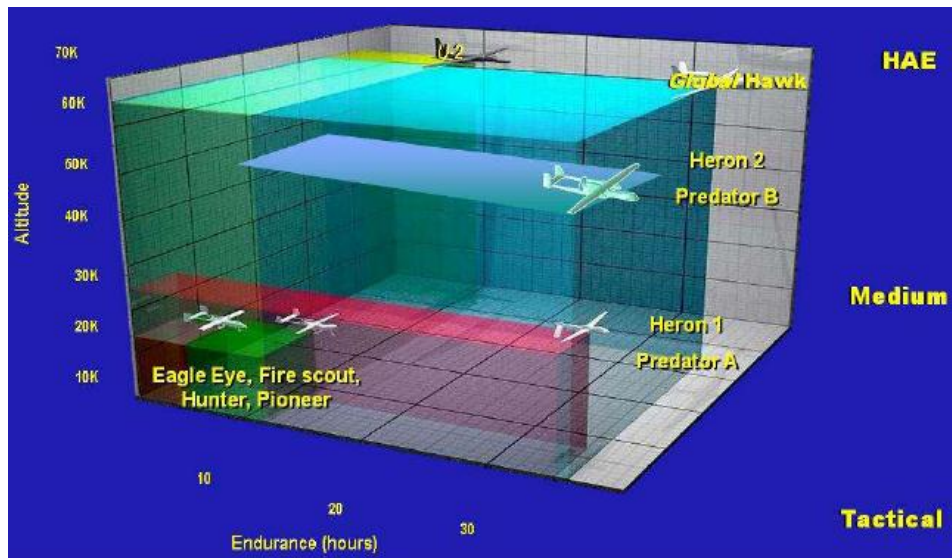


Figura 16. Vehículos no tripulados

El inconveniente que tiene este tipo de vehículo es que tan solo pueden ser usados para misiones estratégicas o militares donde haya un riesgo para los tripulantes. Esto es debido a la mentalidad de la sociedad, puesto que la mayoría no volaría en un avión que no llevase piloto. La seguridad que nos da que haya un piloto para resolver cualquier imprevisto es lo que hace que los UAS nunca sustituyan completamente los vehículos tripulados.

Además, debido al crecimiento de este tipo de avión se han empezado a desarrollar virus contra ellos, e incluso el querer capturarlos para obtener su tecnología.

3.4 Vehículos Espaciales

Los vehículos espaciales son construcciones destinadas a desplazarse en el espacio ultraterrestre o superior, como son, las naves espaciales, los satélites, estaciones espaciales, transbordadores, etc.

Como en los casos anteriores, hay mucha variedad dentro de este ámbito, por lo que el diseño dependerá de la función que vaya a desempeñar dicho vehículo. Eso sí, todos tienen en común dos partes fundamentales: El cohete, que es la parte impulsora compuesta por los motores y el combustible con la finalidad de atravesar la atmósfera terrestre. Y la nave, la cual

será la parte que viaje por el espacio y cuya forma dependerá de su finalidad. Ambas partes deberán ser capaces de aguantar grandes temperaturas y presiones.

Los sistemas de propulsión para los cohetes son de combustible químico, mientras que las naves tienen mayor variedad de motores, pues pueden ser químicos, nucleares, iónicos (se utiliza un haz de iones -moléculas o átomos con carga eléctrica- para la propulsión, habiendo diferentes maneras de acelerar dichos iones según la tecnología usada) o solares. Los cohetes con ese tipo de combustible tienen una duración muy limitada, de forma a su gran gasto de energía en el despegue, por lo que sería imposible ir más allá de Marte. Por ello, los investigadores de la NASA intentan diseñar un sistema auxiliar que ayude a impulsar el cohete y ahorre combustible, de forma que el despegue sea más seguro y se eviten así accidentes.

También se disponen de vehículos no tripulados, la mayoría de ellos son satélites artificiales o naves robóticas dedicadas a la toma de datos o medidas de investigación científica. El último modelo que ha causado gran expectación de este tipo, ha sido la nave Curiosity que consiguió aterrizar con éxito en Marte y tomar datos del planeta.



Figura 17. Simulación del aterrizaje de Curiosity en Marte

Los vehículos espaciales con respecto a los otros tipos de vehículos no han tenido una gran evolución. Sin embargo, en la ciencia ficción podemos encontrar todo tipo de naves espaciales con todo tipo de motores y estructuras imposibles.

3.5 Vehículos Terrestres

Los vehículos terrestres tienen una gran variedad de móviles y tipos. La división que aquí se hace, es una de las múltiples posibilidades que hay.

3.5.1 Vehículos terrestres sobre raíles

El vehículo más importante de esta sección es el tren, el cual es un medio de transporte que circula sobre carriles, compuesto por uno o más vagones o coches arrastrados por una locomotora o bien constituido por automotores. Un coche en este ámbito es un vehículo ferroviario dedicado esencialmente al transporte de viajeros.

Podemos encontrar trenes alimentados de todas las formas posibles: Vapor, diesel, eléctricos o magnéticos. En España predominan los alimentados por corriente eléctrica, pudiendo hacer la siguiente división:

- Baja Potencia: 600, 750 y 1500 V de CC (corriente continua). Es la utilizada por Metros y Tranvías.
- Media Potencia: 1500 y 3000 V de CC ó 15000 y 30000 V de CA (corriente alterna). Es la consumida por los trenes de Renfe y de largo recorrido.
- Altas Prestaciones: 15000 y 30000 V de CA. La tensión requerida por los trenes de alta velocidad.



Figura 18. Diferentes tipos de trenes

El diseño de los trenes, como en todos los casos, dependen de la funcionalidad que tengan. Una de las cosas más importantes a tener en cuenta es el tipo de tracción que requiere el tren, si concentrada (una única locomotora tira del resto de vagones como remolque) o distribuida (se reparte la tracción a lo largo del tren). La tracción distribuida se consigue gracias a los bogies que son el conjunto de dos o tres pares de ruedas, según modelo, montadas sobre sendos ejes próximos, paralelos y solidarios entre sí, que se utilizan en ambos extremos de los vehículos de gran longitud destinados a circular sobre carriles. Pueden ser bogies tractores y o remolcados, estos últimos sin fuerza de tracción.

En España, además de la función no es el único dato a tener en cuenta, sino los recorridos que va a realizar, puesto que en este país se dispone de 4 tipos de ancho de vía:

- 1000mm.- Ancho métrico FEVE
- 1435mm.- Ancho de vía estándar o internacional.
- 1445mm.- Ancho de vía en Metro de Madrid.
- 1668mm.- Ancho ibérico.

Aquellos trenes que han de cruzar las fronteras de nuestro país, es decir, cambiar de ancho de vía, deben de disponer de un sistema especial para poder modificar su ancho sin sufrir descarrilamientos, como es el caso del Talgo.

Pero, a diferencia del resto de vehículos, la parte más importante de los trenes es la infraestructura externa, pues la parte fundamental de su funcionamiento y donde se encuentran la mayoría de los sistemas electrónicos que se ocupan de la seguridad de los pasajeros y se comunican con el tren, aunque en los sistemas de Metro, se están incorporando a bordo la tecnología que hace al tren cada vez más autónomo y seguro.

La infraestructura de los trenes se compone de los raíles, enclavamientos (instalación o sistema de seguridad utilizado para accionar los aparatos de vía y señales garantizando la seguridad en la circulación impidiendo movimientos incompatibles.), balastro (encargado de repartir los esfuerzos del tren y garantizar la estabilidad), la catenaria (hilo portador del cual está suspendido, gracias a unos elementos intermedios, el hilo de contacto del cual el vehículo motor toma la corriente por medio del pantógrafo), etc.

Estos vehículos también disponen de tipos no tripulados, pero al igual que los UAVs, no están muy extendidos debido a la inseguridad que producen en los ciudadanos. Un ejemplo cercano es el tren del aeropuerto de Barajas (figura 19).



Figura 19. Tren no tripulado

3.5.2 Vehículos Terrestres sin raíles

Esta parte conlleva un gran número de vehículos, como los coches, bicicletas, moto, tractor, quad, monopatín, etc. Pero lo que más nos interesan son los automóviles o vehículos de cuatro ruedas con motor.

La definición de automóvil es un vehículo autopropulsado por un motor, cuya finalidad es el transporte terrestre de personas o mercancías sin necesidad de carriles. Así podemos apreciar que el motor es su parte fundamental. Su cometido es transformar la energía eléctrica, calorífica o ambas (depende del tipo que sea) en energía mecánica capaz de realizar un trabajo.

Los diferentes tipos de motores que hay son:

- **Térmicos:** el trabajo lo obtenemos de energía calórica. A su vez se divide en otros dos tipos.
 - *Combustión Interna:* La combustión se debe al fluido motor (la mezcla del comburente-normalmente aire- y el combustible).
 - *Combustión Externa:* La combustión se produce en un fluido que no es el motor.
- **Eléctricos:** el trabajo se obtiene de una corriente eléctrica.

Las características principales que se tienen en cuenta de un motor son la potencia, normalmente dada en caballos (CV) es el trabajo que el motor es capaz de realizar en la unidad de tiempo a una determinada velocidad de giro; Rendimiento, es el cociente entre la potencia útil que generan y la potencia absorbida; Par motor, es el momento de rotación que actúa sobre el eje del motor. Se mide en $\text{kg}\cdot\text{m}$ (kilogramos por metro) o newtons-metro (Nm).

Otro de los elementos, es la dirección, que nos permite modificar la orientación de la trayectoria, ayudándonos a tomar las curvas. En los automóviles la dirección se modifica ante la acción del volante.

La suspensión es el conjunto de elementos que absorben las irregularidades del terreno por el que se circula para aumentar la comodidad y el control del vehículo. Se han desarrollado multitud de suspensiones dependiendo del eje sobre el que actúan, siendo las más famosas son la suspensión MacPherson y la de ballesta (conjunto elástico realizado con láminas de acero). La MacPherson puede ser instalada adelante o atrás. Tiene un solo brazo de control inferior, un ensamble de pierna, amortiguador y un resorte. El brazo de control esta fijo a través de rotulas al chasis y a la parte inferior de la pierna. La parte superior esta sujeta a una sección reforzada de la carrocería. Para apreciarlo mejor, podemos observar la figura 20:

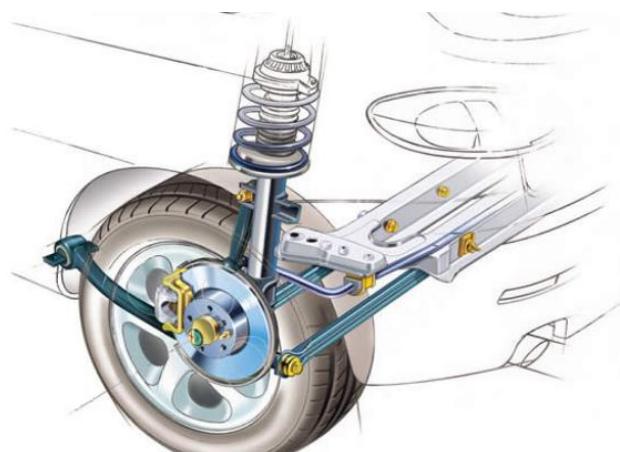


Figura 20. Suspensión MacPherson

Uno de los elementos más importantes de la seguridad son los frenos, que son los encargados de disminuir o parar el movimiento de las ruedas. Su funcionamiento se basa en transformar la energía cinética en térmica. Podemos encontrar diferentes tipos de freno

según el objeto principal que usan para ello, aunque todos ellos funcionan mediante fricción: frenos de cinta, de disco, de tambor o de llanta.

Una vez hemos definido las partes más importantes de los automóviles, veremos los diferentes tipo que hay. Se han desarrollado en todos los ámbitos posibles, desde para uso de la vida cotidiana hasta carreras. Por ejemplo, los todoterreno han sido diseñados para poder enfrentarse, como bien indica su nombre, a cualquier tipo de terreno y tienen la cualidad de tener tracción a las cuatro ruedas y poseer una reductora que permite optimizar la energía del motor, así como ángulos capaces de superar dichos obstáculos. Además, tienen mayor altura que los turismos para poder superar mejor los obstáculos.

Otro vehículo curioso son los buggies, diseñados para poder andar sobre arena, tienen un chasis ligero, una carrocería sin techo rígido, ruedas grandes y la posición del motor en la parte trasera del vehículo.



Figura 21. Automóvil todoterreno y buggy

También se están desarrollando automóviles autónomos. Gracias a una competición realizada en Estados Unidos llamada Grand Challenge, que comenzó en 2004, universidades de todo el mundo presentaban su vehículo autónomo y debían de pasar una serie de pruebas. Los dos primeros años, debían hacer un recorrido en el desierto y el más rápido ganaría. El tercer año se pasó a hacer la prueba en ciudad, donde había más coches (que no eran los participantes) en movimiento y debían superar pruebas de conducción cotidiana. Así, cada participante creaba un método de control que podría desarrollarse posteriormente en los turismos normales. El lado negativo de todo esto, es que la recompensa era cuantiosa por lo que los concursantes no dudaban en elegir el mejor equipamiento y los mejores instrumentos disponibles en el mercado. Esto hace imposible importarlo a los turismos de serie sin que el precio aumente demasiado.



Figura 22. Stanley, ganador Grand Challenge 2005

4. VEHÍCULO DESARROLLADO

4.1. Introducción

En este capítulo se estudiará amplia y detenidamente los pasos seguidos para poder desarrollar el proyecto llevado a cabo. Se explicarán los componentes utilizados, el diseño elegido, cómo se realiza el montaje del mismo y la implementación de la electrónica para el control y el funcionamiento remoto.

4.2. Elementos utilizados

A continuación se enumeran todos los componentes usados, sus características y su finalidad en el proyecto.

4.2.1 Arduino Leonardo

Arduino es una plataforma de electrónica abierta para la creación de prototipos basada en software y hardware OpenSource.



Figura 23. Arduino Leonardo

El hardware utilizado para este proyecto es la tarjeta Leonardo (figura 23), la última en salir al mercado con las siguientes características:

La parte más importante es el microcontrolador ATmega32u (Véase *anexo A.4*), puesto que gracias a él contamos con 20 pines de I/O (entrada/salida) digitales y a su vez podemos usar 7 de ellos como salidas PWM (Modulación por Ancho de Pulso) y 12 como analógicas. Además, cuenta con un cristal de 16 MHz, conexión micro USB, conector de alimentación, conector de ICSP (In-Circuit Serial Programming) y un botón para reiniciar.

Las novedades que incorpora Leonardo respecto a las otras placas es que el nuevo microprocesador cuenta con comunicación USB, mientras que las otras requerían de un procesador secundario. Es decir, podemos controlar el teclado y el ratón desde Arduino con unas librerías especiales que podemos encontrar en la página oficial de Arduino en inglés (la página española aún no está actualizada).

La placa puede ser alimentada mediante el puerto micro USB o una fuente externa. Ésta última puede ser conectada a la placa mediante un adaptador jack de 2.1 mm con centro positivo o ir directamente a los pines Vin y GND. En este proyecto se ha optado por usar como fuente externa una *batería* (véase apartado 4.2.4) enchufada al adaptador jack.

Los valores de alimentación que debe tener la fuente deben estar entre los 6 y 20 V, puesto que si no llegamos a los 6V, el pin que tiene arduino para dar 5V no es seguro que llegue a dar ese valor y la placa puede volverse inestable. Y si se pasa de los 20 V podemos sobrecalentar la placa o estropearla.

La siguiente característica es la memoria de 32KB que posee, aunque 4KB son utilizados para el bootloader (o gestor de arranque). También incorpora 2,5 KB de SRAM (Static Random Access Memory es una memoria capaz de mantener los datos mientras esté alimentada. Pero los pierde una vez se deja de alimentar) y 1 KB de EEPROM (Electrically Erasable Programmable Read-Only Memory, es decir, un tipo de memoria ROM que puede ser programada, borrada y reprogramada eléctricamente).

Voy a hablar más específicamente de alguno de los pines de la placa. Como ya se ha mencionado posee 20 pines digitales que darán un máximo de 40 mA y pueden tener varias funciones:

- *Serial*: Los pines 0 y 1 actúan como recepción (Rx) y transmisión (Tx) de datos respectivamente. Según la comunicación que usemos tenemos las librerías *Serial* para la comunicación USB y *Serial1* para serie TTL (transistor-transistor logic). Además, esta placa incluye la novedad de que si le incorporas al software la librería *SoftwareSerial* (creada para el microcontrolador que lleva la placa Leonardo) puedes usar cualquier pin digital como receptor o transmisor.
- *Interrupciones Externas*: Los pines 2 y 3 pueden emplearse para actuar como interrupción ante flancos o cambios de valor.
- *PWM*: Son los pines 3, 5, 6, 9, 10, 11 y 13. Aportan 8 bits de salida PWM.
- *LED*: Hay un led conectado al pin 13 que se muestra en la placa junto a los leds que indican que está encendida y la comunicación por USB.
- *Entradas Analógicas*: Se disponen de 12 entradas analógicas que están separadas en dos grupos, del A0-A5 están en la parte derecha de la placa, mientras que las entradas A6-A11 se encuentran en los pines digitales 4,6,8,9,10 y 12. Cada entrada puede proporcionar 10 bits de resolución y la medida va de 0 a 5 V, aunque puede modificarse.

Un componente importante que incorpora esta tarjeta es un polifusible reseteable que protege los puertos USB del ordenador ante variaciones bruscas y altas de corriente. Aunque es una protección extra, puesto que los ordenadores ya suelen tener una protección interna para estos casos. Si llegamos a aplicar más de 500 mA al puerto USB este elemento saltará y cortará la conexión hasta que el cortocircuito o sobrecarga desaparezcan.

Las características físicas de la placa son de 2,7 pulgadas de largo (6,9 cm) y 2,1 de ancho (5,3 cm). Posee cuatro orificios para los tornillos que nos permiten sujetar la placa a una superficie. La distancia entre pines es de 100 milésimas de pulgada y de 160 milésimas entre los pines 7 y 8.

Para programar la placa se ha usado el software Arduino 1.0.1, el cual es la última versión hasta el momento del programa. En él se usa un editor de texto para escribir el código, un área de mensajes para mostrar los errores de compilación o el correcto funcionamiento de la tarea encomendada, una barra de herramientas con botones para las funciones comunes como compilar/cargar/guardar etc. Y en la parte superior una serie de menús donde podremos encontrar las opciones del puerto al que está conectada nuestra placa, elegir el nombre de la placa que estamos usando entre la lista de las diferentes arduino, etc. En la figura 24 se puede apreciar más claramente las partes mencionadas:

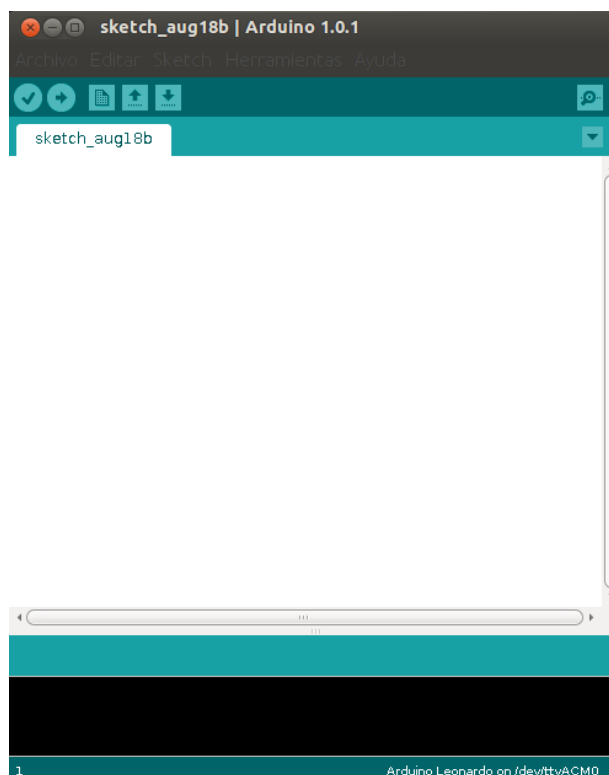


Figura 24. Entorno Arduino 1.0.1

4.2.2 Servomotores

Los servomotores o servos son motores de corriente continua que constan con la capacidad de ser controlados tanto en velocidad como en posición. Esto se debe a que están formados por un motor, una caja reductora y un circuito de control. Éste último está formado por un circuito electrónico y un potenciómetro que indica la posición exacta del servo.



Figura 25. Partes de un servo

El control de los motores se lleva a cabo mediante la modulación por ancho de pulso o PWM. La mayoría trabajan en la frecuencia de 50 Hz, por lo que las señales de PWM tendrán un periodo de 20 ms. El circuito de control responderá al ancho de la señal modulada de la siguiente manera: Si el ancho de la señal recibida está entre 0,3 y 1,4 ms el motor se moverá en sentido horario; entre 1,6 y 2 ms se moverá en sentido antihorario; y finalmente, si reciben 1,5 ms se mantendrán en reposo. Podemos apreciarlo mejor en la siguiente figura:

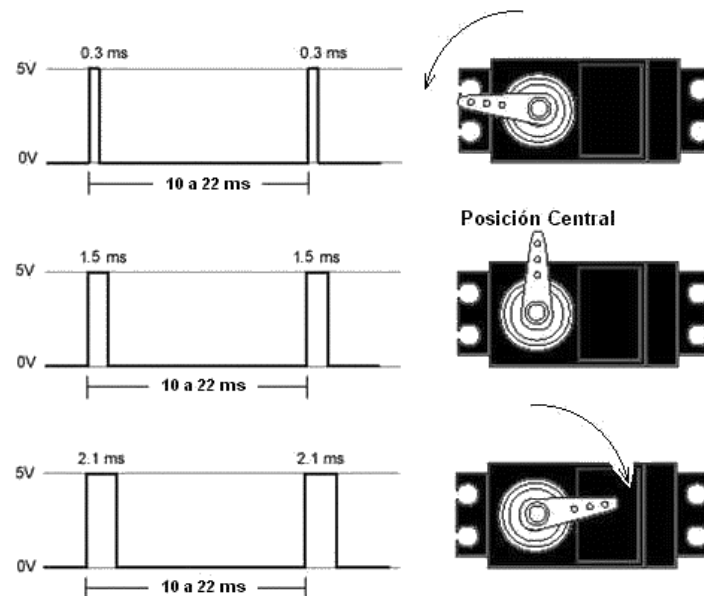


Figura 26. Señal y posición del servo

Los servos reciben las señales por tres cables: el rojo para la alimentación del motor y los componentes de control con valores entre 4.8 y 6 V; el negro es el cable de tierra para los elementos ya mencionados; y un cable blanco o amarillo (depende del servo) que envía la señal de control para determinar la posición y movimiento.

El modelo de servo usado para este proyecto ha sido **Futaba s3003** (figura 27) cuya finalidad es otorgar dirección al vehículo. Sus dimensiones son 41 x 20 x 37 mm y un peso total de 43 gr. Sus características principales a 4.8 V son:

- Velocidad: 0.23 s/60 grados
- Torque (par motor): 3.2 Kg/cm
- Posición: Entre 0 y 180°



Figura 27. Servo Futaba s3003

4.2.3 Motores DC

Los motores de corriente continua son máquinas que convierten la energía eléctrica en mecánica, produciendo un movimiento rotatorio. Este tipo de motores han sido los más usados en la industria por su facilidad de controlar posición y velocidad, pero con los avances de la electrónica están siendo sustituidos por motores de corriente alterna, asíncronos. Esto ha sido debido a que su coste y mantenimiento es más barato que los motores de continua.

Un motor de corriente continua se compone de un estator, el cual se encarga de dar soporte mecánico al aparato y es el lugar donde se encuentran los polos (imanes permanentes o devanados con hilo de cobre sobre núcleo de hierro). Y la otra parte es el rotor, el cual suele tener forma cilíndrica, devanado y un núcleo al que le llega la corriente mediante dos escobillas.

Su funcionamiento se basa en la *Ley de Lorentz*: cuando un conductor por el que pasa la corriente eléctrica se sumerge en un campo magnético, el conductor sufre una fuerza perpendicular al plano formado por el campo magnético y la corriente, siguiendo la regla de la mano izquierda (Para obtener el sentido de la fuerza, se toma el dedo índice de la mano apuntando a la dirección del campo magnético que interactúa con el conductor y con el dedo corazón se apunta en dirección a la corriente que circula por el conductor, formando un ángulo de 90 grados. De esta manera, el dedo pulgar determina el sentido de la fuerza que experimentará ese conductor). La fuerza tendrá un módulo igual a: $F = B \cdot l \cdot I$ siendo B la densidad del campo magnético, l la longitud del conductor (en metros) e I la intensidad que recorre el conductor (en amperios). En el rotor se encuentran varios conductores en la periferia, de forma que según gira, se activa el conductor adecuado.

El sentido de giro de un motor DC (Direct Current) depende del sentido de la corriente que circula por los devanados inductor e inducido. Para invertir el sentido de giro, tan solo debemos cambiar el sentido del campo magnético o la corriente del inducido. El sentido de giro lo podemos determinar con la regla de la mano derecha (el pulgar apunta la dirección de la corriente mientras los demás dedos declaran la rotación natural del campo magnético), la cual nos va a mostrar el sentido de la fuerza. En la figura 28 podemos apreciar los pasos que hace el motor para realizar el giro, viendo mejor todo lo explicado.

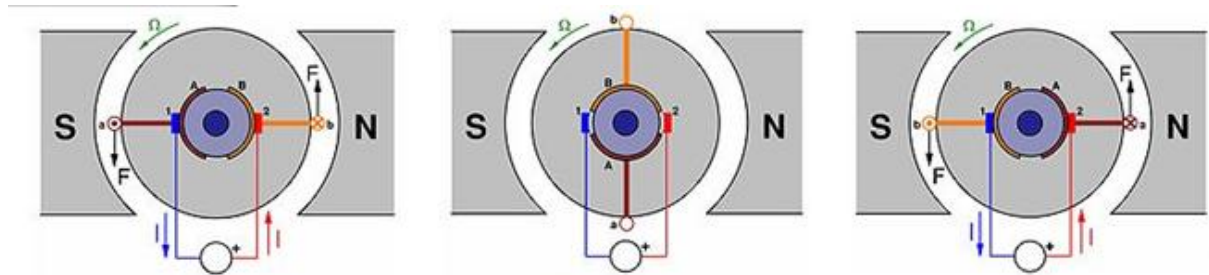


Figura 28. Funcionamiento de un motor DC

Una vez explicado el funcionamiento general de los motores de continua pasamos a especificar los usados en este proyecto. Las máquinas utilizadas en este caso han sido rescatadas de un antiguo coche de radio control, por lo que encontrar el modelo exacto ha sido complicado, de forma que se ha buscado el equivalente en la actualidad. Así, tras una larga búsqueda se ha llegado a la conclusión de que el modelo equivalente es el motor Mabuchi (misma marca que los antiguos) RS-540SH-6527 (véase *anexo A.1*). Han sido diseñados especialmente para uso de ocio, es decir, juguetes y radio control. Sus principales características son:

- Rango de tensión: 4,8- 9,6 V
- Velocidad a la máxima eficiencia: 20040 rpm
- Torque a la máxima eficiencia: 31 mN·m

De esta forma podemos apreciar que este motor es muy rápido pero el torque o par motor es muy pequeño, por lo que tendremos que acompañar de un sistema reductor a cada motor si queremos que puedan hacer andar el vehículo, con y sin la carga del brazo robótico.

4.2.4 Batería LIPO

Una batería es un dispositivo que almacena energía eléctrica usando procedimientos electroquímicos, que posibilitan la recarga casi en su totalidad, pudiendo repetir este ciclo un número limitado de veces. Existen una gran variedad de tipos, dependiendo de la finalidad. En nuestro caso, usamos una batería de polímero de litio (LIPO).

Las LIPO son una variación de las baterías de iones de litio (Li-ion). Tienen características similares, pero las de polímero de litio tienen un tamaño y peso más reducido y una mayor densidad de energía. Son muy útiles para el radio control, puesto que ofrecen potencia y duración.

Hay cuatro tipos de esta batería:

- 1S: una celda → 3,7 V
- 2S: dos celdas → 7,4 V
- 3S: tres celdas → 11,1 V
- 4S: cuatro celdas → 14,8 V

Como podemos observar, cada celda tiene un voltaje de 3,7 V permitiendo hasta un máximo de 4,2 V y un mínimo de 3 V. Es muy importante no dejar que caiga del mínimo, puesto que entonces la batería se daña de forma irreparable. Además, no se pueden exponer a altas temperaturas, ni recibir fuertes golpes, ni podemos dejar que se toquen los dos terminales, puesto que se puede correr el riesgo de que se incendie.

Para el vehículo se ha elegido una batería LIPO de la marca Zippy de 2S (7.4 V) con una capacidad de 1000 mAh como la mostrada en la ilustración 29. El motivo por el que se ha escogido con esas características ha sido porque es pequeña, tiene el valor medio de tensión adecuado tanto para la placa Arduino como para el motor DC y con 1000mah garantizamos una larga duración de funcionamiento. Sus dimensiones son 71x35x11mm.



Figura 29. Batería LIPO Zippy

4.2.5 Rodamientos

Los rodamientos son un elemento mecánico que reduce la fricción entre un eje y las piezas conectadas a éste por medio de rodadura, que le sirve de apoyo y facilita su desplazamiento. El movimiento rotativo puede ser axial (soporta esfuerzo en la dirección de su eje), radial (soporta esfuerzos de dirección perpendicular a la dirección que pasa por el centro de su eje) o axial-radial (si es capaz de soportar ambos esfuerzos).

Hay varios tipos de rodamientos y cada uno muestra unas características según el diseño, que lo hacen adecuado para unas aplicaciones determinadas. Por ejemplo, los **rodamientos rígidos de bolas** pueden soportar los dos tipos de esfuerzos antes mencionados de forma moderada y poseen baja fricción, por lo que son adecuados para motores eléctricos de medio y pequeño tamaño. Por ello han sido escogidos para usarse en este proyecto.



Figura 30. Rodamientos rígidos de bola

Además de las propiedades ya mencionadas, son fáciles de diseñar, no separables, no necesitan un gran mantenimiento y son capaces de operar a altas velocidades. Todo ello con su módico precio hacen de estos rodamientos los más famosos y utilizados.

El vehículo lleva rodamientos en las ruedas delanteras puesto que son pasivas, es decir, no tienen tracción, y el rodamiento es usado como punto de unión de la rueda con el resto del vehículo.

4.2.6 Circuito Integrado L293B

Un puente H es un circuito electrónico que permite que un motor gire en ambos sentidos. Un esquema básico de este circuito es la figura 31.

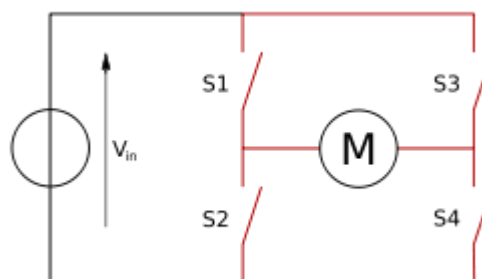


Figura 31. Esquema Puente H

Así su funcionamiento se basa en el control de los interruptores o transistores (llamados S_x en la representación) que dispone, de forma que podemos encontrar cuatro estados distintos del circuito:

- S_1 y S_4 abiertos (S_2 y S_3 cerrados): se aplica una tensión positiva en el motor de modo que comienza a girar en uno de los sentidos.
- S_2 y S_3 abiertos: se aplica una tensión negativa en el motor, por lo que el sentido de la corriente ha sido modificado y ahora el motor girará en el otro sentido.
- S_1 y S_3 abiertos o S_2 y S_4 abiertos: El motor se frena debido a la fuerza electromotriz generada (diferencia de potencial entre dos puntos de un circuito abierto o de producir una corriente eléctrica en un circuito cerrado).

En ningún caso podemos permitir que se activen a la vez los interruptores S_1 y S_2 o S_3 y S_4 a la vez, puesto que provocarían un cortocircuito en la fuente de tensión.

En conclusión, para poder mover un motor en ambos sentidos necesitamos un circuito en puente H. De esta forma podemos encontrar varios tipos de circuitos integrados con este esquemático, pero el elegido en este caso ha sido el modelo L293B (véase Anexo A.3). Las principales características de este modelo son las siguientes:

- Tensión de alimentación (V_s) máxima: 36 V
- Tensión de alimentación lógica (V_{ss}) máxima: 36 V
- Pico de corriente de salida (I_{out}): 2A
- Rango de temperatura: -40 a 150°C



Figura 32. Circuito integrado L293B

4.2.7 Control Remoto: GamePad

Un gamepad es un dispositivo de entrada usado para interactuar con un videojuego, permitiendo moverse e interactuar con los elementos del juego para conseguir los objetivos del mismo.

En este caso lo usaremos para controlar nuestro vehículo mediante programación C++. El modelo de gamepad escogido ha sido el mando Xbox 360 por su precisión, diseño ergonómico y la posibilidad de contar con 2 gatillos analógicos. Características:

- Botón central luminoso
- 8 botones de juego, 2 joysticks, 2 gatillos electrónicos y una cruceta.
- Comunicación USB



Figura 33. Mando Xbox 360

4.2.8 Módulo Bluetooth

Bluetooth es una red inalámbrica de área personal que permite la transmisión de voz y datos entre diferentes dispositivos mediante un enlace por radiofrecuencia de 2,4 GHz. Gracias a esta red facilitamos las comunicaciones entre equipos móviles y fijos, y eliminamos cableado.

Bluetooth también se llama al protocolo de comunicaciones diseñado especialmente para dispositivos de bajo consumo y con cobertura baja. Con él, podemos comunicar varios dispositivos dentro de lo que la potencia de transmisión nos permita. Los dispositivos son clasificados de la siguiente manera:

- *Clase 1:* Tiene una potencia máxima de 20 dBm (nivel de potencia en decibelios en relación a un nivel de referencia de 1 mW), lo que le permite alcanzar un rango de unos 100 metros.
- *Clase 2:* Potencia máxima de 6 dBm, consiguiendo un alcance de unos 10 m.
- *Clase 3:* Potencia máxima de 1 dBm, teniendo un rango de 1 m.

Todas las clases son compatibles entre sí. Incluso si conectamos uno de clase 2 con otro de clase 1, podemos conseguir que el primero tenga un rango mayor debido a la alta potencia de transmisión del segundo.

Por otro lado, según la versión que sea tendremos diferentes anchos de banda teniendo un mínimo de 1 Mbit/s (versión 1.2) hasta 24 Mbit/s (versión 4.0).

Para la comunicación inalámbrica del proyecto se ha elegido el **módulo bluetooth JY-MCU** para arduino. Sus características principales son:

- Protocolo Bluetooth Versión 2.0
- Clase 2 (+6dBm)
- Sensibilidad receptora: -85 dBm
- Rango de tensión de alimentación: 3,3 – 5V.
- Rango de temperatura: -40 a +150°C
- Referencia de energía: Buscando dispositivo 35 mA, conectado 8 mA.
- Comunicación con la placa arduino mediante puerto serie RS232 (nivel TTL).

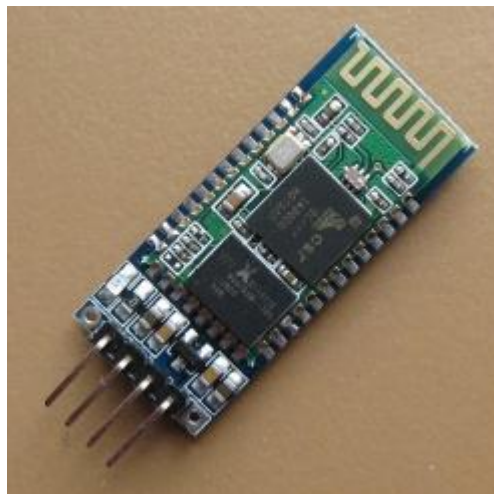


Figura 34. Módulo Bluetooth JY-MCU

4.2.9 Metacrilato

Metacrilato es el nombre con el que se conocen comúnmente las planchas o láminas de polimetacrilato. Éste compuesto se obtiene de la polimerización del metacrilato de metilo (compuesto en estado líquido y tóxico) y las representaciones más comunes en la industria son en gránulos, láminas, tubos o varillas. A continuación se detallan las principales propiedades de este material:

- Transparencia de un 93%.
- Resistencia al impacto de diez a veinte veces mayor que el vidrio.
- Resistente a la intemperie y a los rayos ultravioletas.
- Buen aislante térmico y acústico.
- Componente ligero con una densidad de 1190 kg/m³.
- Se raya fácilmente con cualquier objeto metálico.
- De fácil combustión, no se apaga al ser retirado del fuego. No produce gases tóxicos al arder.
- Facilidad de mecanizado y moldeo

- Se puede mecanizar en frío pero no doblar. Proceso complicado en el que se debería calentar la zona a doblar o toda la pieza.

Se ha usado este componente para realizar la base del vehículo, teniendo unas dimensiones de 20 x 13 x 0,2 cm.

4.3. Diseño

Una vez numerados y explicados los elementos más importantes que intervienen en el proyecto, el siguiente paso es razonar el diseño elegido para el vehículo.

4.3.1 Modelo seleccionado

En la actualidad hay seis tipos de robots móviles:

- Diferencial (1)
- Triciclo (2)
- Síncrono (3)
- Automóvil o Ackerman(4)
- Omnidireccional (5)
- Robot con patas(6)

Siendo los primeros cinco movidos mediante ruedas.

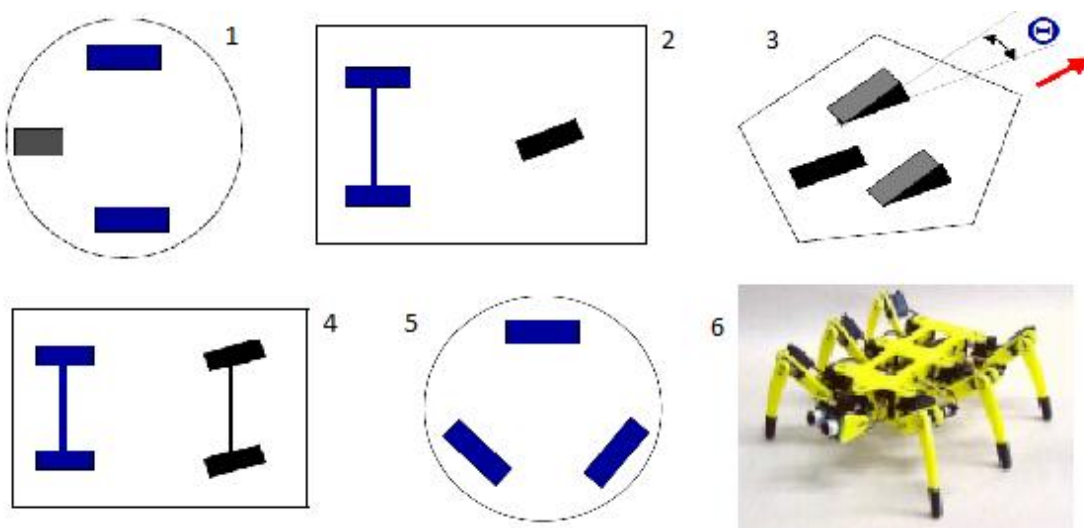


Figura 35. Tipos de robots móviles

El **modo diferencial** tiene uno de los esquemas más sencillos que podemos encontrar dado que está compuesto por dos ruedas paralelas entre sí y una (configuración en triángulo) o dos (configuración diamante) más auxiliares para conseguir estabilidad. El nombre de este tipo de robots viene dado porque la dirección del mismo se debe a la diferencia de velocidades

entre las ruedas que se encuentran paralelas entre sí, puesto que también son las que llevan la tracción. Pueden moverse recto (misma velocidad en las dos ruedas), en arco o sobre su propio eje.

El **modo triciclo** tiene un esquema parecido al anterior porque está compuesto por dos ruedas paralelas entre sí que dan la tracción al robot y una tercera que da la dirección. Al ser la tercera la que da dirección hace que la cinemática del mismo sea más complicada que la anterior y limitando los movimientos a recto y en arco.

El **modo síncrono** se basa en que las ruedas se mueven de forma síncrona, es decir, a la vez. Las ruedas siempre miran en una dirección y para girar, rotan sobre el eje vertical. Para ver este movimiento más claro véase la figura 35. Este es un tipo especial del modo diferencial.

El **modo omnidireccional** tiene tres ruedas colocadas con un ángulo de 120° entre sí. Cada rueda tiene la capacidad de girar en ambos sentidos gracias a lo cual se consigue un control lineal más sencillo que el modo diferencial.

Los **robots con patas**, al no utilizar ruedas, tienen mayor dificultad para mantenerse estables, además de que requieren de mayor consumo de energía para contrarrestar la gravedad. Su control es más complejo que cualquiera de los mencionados anteriormente, pero son los más llamativos y los que causan mayor impresión en la gente.

El **modo automóvil** o modelo Ackerman es una evolución del modo triciclo puesto que utiliza dos ruedas para la tracción y otras dos para la dirección, aunque tiene mayor complejidad cinemática y mecánica que él. Este modo destaca por su estabilidad y su capacidad de ir recto. Éste ha sido el modo seleccionado para el proyecto debido a que uno de los objetivos es que pueda llevar un brazo robótico encima, por lo que he considerado la **estabilidad** uno de los puntos más importantes para elegir modelo. De esta forma evitamos que el coche pueda volcar al transportar el brazo.

4.3.2 Cinemática

En primer lugar vamos a explicar dos tipos de restricciones cinemáticas relacionadas con la movilidad del robot:

- **Holonómicas:** los diferentes grados de libertad están desacoplados, permitiendo movimientos como el de rotación sobre su propio eje. De este tipo son los robots diferenciales y síncronos.
- **No-Holonómicas:** los grados de libertad están acoplados, es decir, para dar la vuelta el robot ha de ir hacia delante o hacia atrás mientras gira. Es el caso de los robots triciclo y automóvil.

Así podemos apreciar que el vehículo va a tener restricción no-holonómica, por lo que para moverse y llegar al punto deseado deberá realizar más maniobras que un robot holonómico.

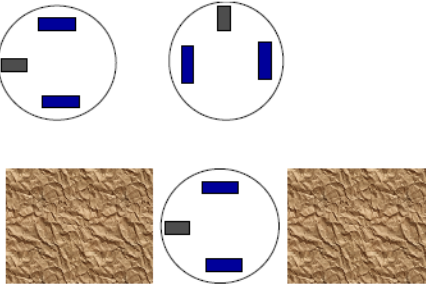
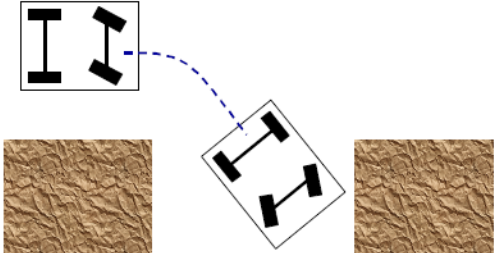
Holonómicas -- DIFERENCIAL --	No-holonómicas --AUTOMÓVIL--
	

Figura 36. Restricciones cinemáticas

Ahora pasamos a estudiar la cinemática de nuestro modelo. En primer lugar, tenemos dos sistemas de coordenadas: el sistema fijo llamado "O" y un sistema móvil situado en el vehículo, llamado "M". Podemos verlos representados en la figura 37:

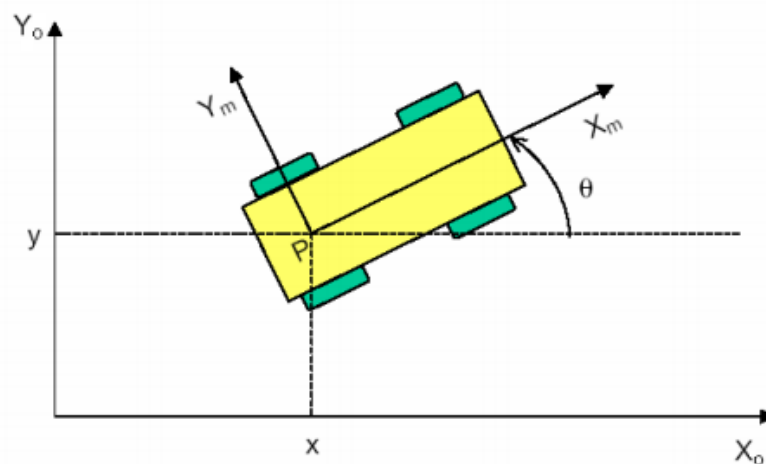


Figura 37. Sistemas de coordenadas

Para poder pasar de un sistema a otro necesitaremos una matriz de transformación entre ambas, tomando P como punto de referencia:

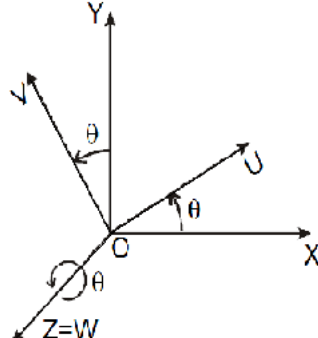
$${}^O P = {}^O T_M \cdot {}^M P$$

$${}^M P = {}^O T_M^{-1} \cdot {}^O P$$

Por lo que analizando la representación de la figura 37 podemos decir que la matriz de transformación T es la siguiente, sabiendo que dicha matriz está formada a su vez por otras cuatro:

$$T = \begin{bmatrix} R_{3 \times 3} & P_{3 \times 1} \\ f_{1 \times 3} & E_{1 \times 1} \end{bmatrix} = \begin{bmatrix} \text{Matriz de rotación} & \text{Vector de posición} \\ \text{Transformación de perspectiva} & \text{Escalado global} \end{bmatrix}$$

Siendo la matriz de rotación (en este caso del eje z) la siguiente:

$$R(z, \vartheta) = \begin{bmatrix} \cos \vartheta & -\sin \vartheta & 0 \\ \sin \vartheta & \cos \vartheta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$


Por lo que la matriz de transformación queda así:

$${}^o_M T = \begin{bmatrix} \cos \theta & -\sin \theta & 0 & x \\ \sin \theta & \cos \theta & 0 & y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Para desarrollar el modelo cinemático del modelo consideraremos las siguientes hipótesis:

- El robot se mueve en superficie plana
- Ejes de guiado perpendiculares al suelo
- Movimiento con rodadura pura
- El robot se comporta como un sólido rígido
- El movimiento se realiza alrededor de un arco de circunferencia

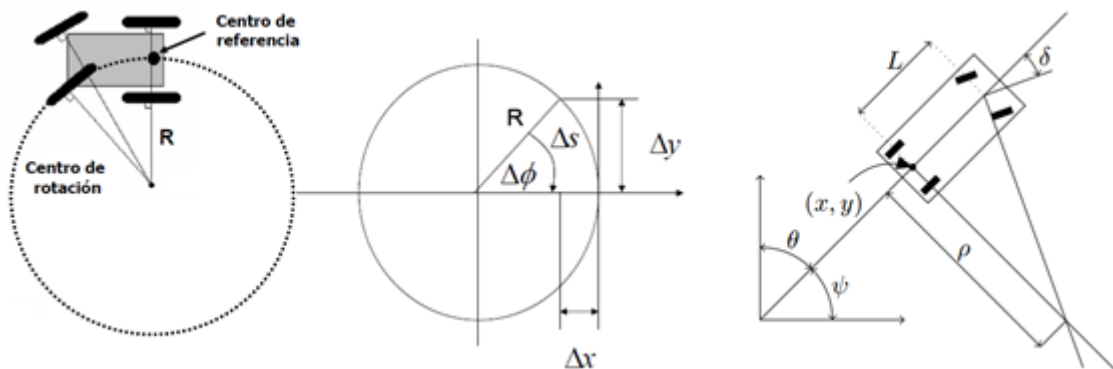


Figura 38. Análisis cinemático

Por lo que obtenemos que $\Delta S = v$ (velocidad lineal dada por las ruedas traseras) y $\Delta\Phi = w$ (velocidad angular de las ruedas de dirección). Si observamos la tercera representación empezando por la izquierda de la figura 38, podemos observar el modelo cinemático basado en el modelo de Ackerman, donde x e y son nuestro punto de referencia, L la distancia entre los ejes delantero y trasero del coche, δ el ángulo que rotan las ruedas delanteras, ψ el ángulo que está rotado el sistema móvil del coche y ρ el radio de la circunferencia que dibujan las ruedas de dirección al girar.

Comenzaremos proyectando el vector velocidad (tangente a la circunferencia de giro) sobre los ejes del sistema fijo:

$x = V \cdot \cos\psi$ siendo V a su vez $V = V_s \cdot \cos\delta$ siendo V_s la velocidad en el sistema global:

$$y = V \cdot \sin\psi$$

$$\psi = w$$

Además, $V_s = W_s \cdot R$ siendo R el radio de la rueda de tracción.

Con estas ecuaciones podemos hallar la matriz Jacobiana que nos permitirá hallar la cinemática directa:

$$J = \begin{bmatrix} \frac{\partial x}{\partial v} & \frac{\partial x}{\partial w} \\ \frac{\partial y}{\partial v} & \frac{\partial y}{\partial w} \\ \frac{\partial \psi}{\partial v} & \frac{\partial \psi}{\partial w} \end{bmatrix} = \begin{bmatrix} \cos\psi & 0 \\ \sin\psi & 0 \\ 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \end{bmatrix} = J \cdot \begin{bmatrix} V \\ w \end{bmatrix}$$

Por lo que obtenemos nuestro modelo cinemático.

$$\dot{x} = V \cdot \cos\psi = V \cdot \cos\left(\frac{\pi}{2} - \theta\right)$$

$$\dot{y} = V \cdot \sin\psi = V \cdot \sin\left(\frac{\pi}{2} - \theta\right)$$

$$\dot{\psi} = w = \frac{V}{\rho} = \frac{V}{L} \cdot \tan\delta \quad \text{Debido a} \quad \rho = \frac{L}{\tan\delta}$$

$\dot{\delta} = w$ (Se obtiene de hallar la derivada del ángulo del giro, dando como resultado velocidad angular).

Además, hemos de añadir la ecuación dada por la restricción no-holonómica:

$$\dot{x} \cdot \sin\theta - \dot{y} \cdot \cos\theta = 0$$

4.3.3 Suspensión

Como ya se explicó anteriormente, la suspensión es el conjunto de elementos que absorben las irregularidades del terreno por el que se circula para aumentar la comodidad y el control del vehículo.

El coche está compuesto de dos sistemas de suspensión pasivos (es decir, no tienen un sistema de control que según la situación activen actuadores para mantener la estabilidad): uno delantero, mediante muelles y otro trasero, mediante ruedas que se doblan para adaptarse al terreno o al peso.

El primero de ellos fue inspirado en los coches de radio control. Tiene una inclinación de unos 60° , menor que la mayoría de radio control o más pequeña que los que tienen esa inclinación. Esto se debe a que con el diseño realizado de la parte delantera, ese ángulo permite gran movilidad a la rueda para adaptarse al terreno.



Figura 39. Comparación de suspensión

La suspensión delantera es el resultado de la suspensión realizada por Daniel Gómez (subida en la página web de Thingiverse: <http://www.thingiverse.com/thing:14602>) con las modificaciones necesarias para que encajara con mi modelo. Con este diseño el muelle que encajaba era el de métrica 8 y se implementó uno blando (fácil flexión), puesto que el duro requería de un gran esfuerzo para que se flexionara y la presión que hacía sobre las piezas era peligroso para éstas. Para encajar las dos piezas que forman la suspensión se utilizó goma, puesto que es flexible y no interfiere en el movimiento del muelle, mientras que otros materiales entorpecían los elementos haciendo que se quedara encajada.

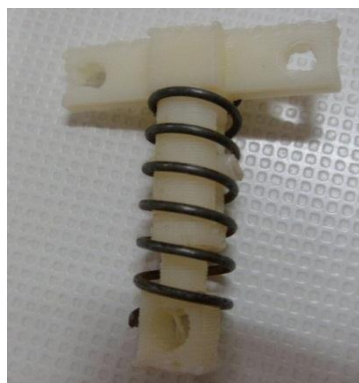


Figura 40. Suspensión realizada por Daniel Gómez

La suspensión trasera fue bajada directamente de la página web antes mencionada (<http://www.thingiverse.com/thing:17514>) siendo su creador el usuario "tjhowse". A partir de ella se han hecho varias evoluciones y se han parametrizado para elegir el número de radiales que se necesitaban. En nuestro caso, escogí el modelo original de seis radiales, puesto que tras probarla la creí la más apropiada para el peso que ha de soportar el vehículo.

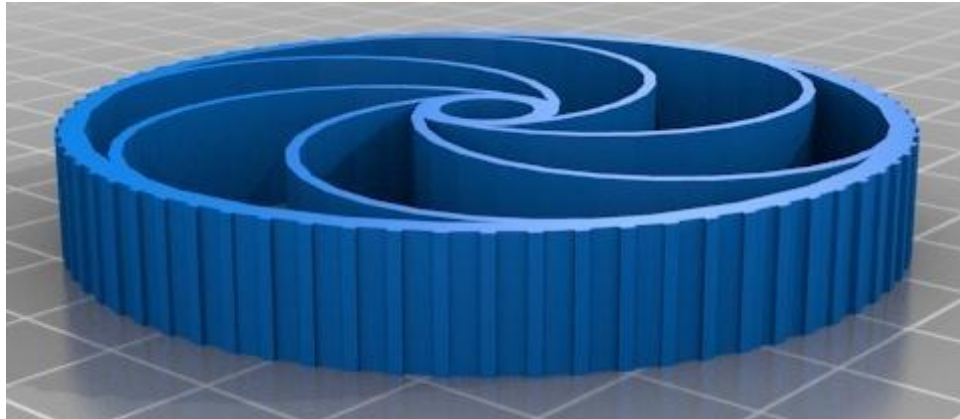


Figura 41. Parametric airless tire

De esta forma, aumentamos la estabilidad que ya proporciona el coche para cualquier carga que necesite llevar y lo habilitamos para terrenos más abruptos.

4.3.4 Dirección

La dirección nos permite modificar la orientación de la trayectoria. En nuestro vehículo será controlada mediante un servo.

El diseño de la dirección, como en la suspensión, también está inspirado en los coches radio control, pero guarda menos parecido que la suspensión. Fue más difícil elaborarla porque debía tener en cuenta todo el diseño que llevaba hecho de la parte delantera para que las diferentes partes no se molestasen entre sí y, además, aproveché para que el servo sirviera de unión entre la parte central del coche y la parte delantera.

Finalmente, el diseño fue el siguiente: En las ruedas delanteras, en la parte central se les acopló unos rodamientos, puesto que eran ruedas pasivas, y el centro de éstos se unió con la pieza base de dirección. Ésta última también funciona como intermediaria entre las piezas con suspensión y la rueda.

El servo se colocó en el centro de la base del coche produciendo el inconveniente de que la parte que proporciona el movimiento no estaba centrada. La parte con más problemas fue la unión servo-pieza de dirección. Las opciones de hacer la unión mediante un sistema de cremallera no eran factibles puesto que la unión del servo con la parte delantera del coche impedía cualquier idea de ese tipo, además de que las piezas no se encontraban a la misma altura.

Se probó imprimiendo piezas de las medidas adecuadas para cada lado y que “tirasen” o “empujasen” la dirección, pero no fue posible puesto que el movimiento que hacían las piezas era demasiado holgado, perdiendo mucho giro. Y por el contrario, si se intentaba “engordar” la pieza para evitar esas holguras, se encajaba de tal forma que no permitía bien el movimiento.

La segunda idea fue usar bridas como unión, puesto que podría ajustarlas lo que necesitara para evitar holguras. El problema resultó que la holgura fue mayor que el caso anterior, debido a que los ajustes que hacía a su vez con otras bridas resbalaban entre sí.

La idea que finalmente se llevó a cabo fue la de usar alambre con cierta rigidez para evitar que se flexionara o amoldase con cada movimiento. De forma que se entrelazó el alambre para evitar dobleces y se ajustó para que la holgura que tuviera fuera mínima.

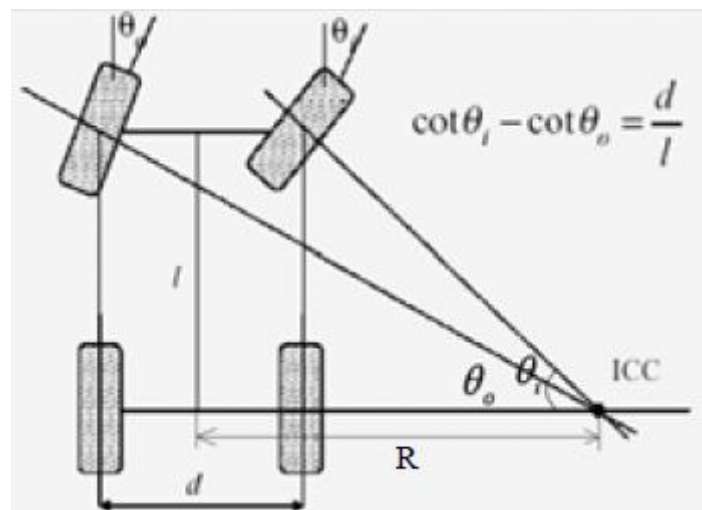


Figura 42. Modelo de dirección Ackerman

A parte, en este tipo de modelo se suelen configurar las ruedas de dirección de forma que la interior a la curva gire más que la exterior, para reducir deslizamientos (véase figura 42). Para saber si el ángulo de cada rueda es el adecuado, se ha de cumplir la siguiente ecuación:

$$\cotg \theta_i - \cotg \theta_o = \frac{d}{l}$$

Siendo:

- d la separación lateral de las ruedas
- l separación longitudinal de las ruedas
- θ_i ángulo relativo de dirección de la rueda interior
- θ_o ángulo relativo de dirección de la rueda exterior

En este caso, los datos obtenidos tras el montaje son los siguientes:

$$d = 16 \text{ cm} \quad l = 21 \text{ cm} \quad \theta_i = 25^\circ \quad \theta_o = 20^\circ$$

Por lo que calculando ambos lados de la ecuación, obtenemos los siguientes resultados:

$$\cotg \theta_i - \cotg \theta_o = \cotg 25 - \cotg 20 = \mathbf{0.56}$$

$$\frac{d}{l} = \frac{16}{21} = \mathbf{0.76}$$

Podemos apreciar que los resultados no coinciden. Para que lo hicieran se calculó que la diferencia entre los ángulos de las ruedas deberían ser de dos grados más, es decir:

$$\cotg \theta_i - \cotg \theta_o = \cotg 27 - \cotg 20 = \mathbf{0.74}$$

Es un resultado mucho más aproximado que el que teníamos. Se intentó llevar a cabo esta última configuración, pero siendo el alambre colocado manualmente, el mejor resultado obtenido fue el anterior.

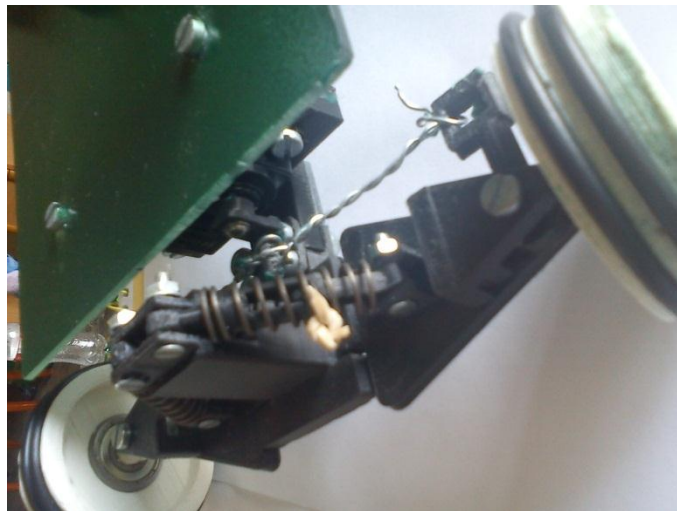


Figura 43. Sistema de dirección con alambre

4.3.5 Reductor

El vehículo que se ha desarrollado se mueve gracias a motores de corriente continua, con la peculiaridad de que no se usa un único motor con un solo eje en la parte trasera, sino que se usa un motor para cada rueda. Esto fue debido a que inicialmente usaba unos motores mucho más pequeños y menos potentes que los que están puestos, por lo que pensando en la carga que podría soportar, con uno solo de ellos no iba a bastar. Por esta razón la idea inicial fue la de que cada rueda llevase un motor consigo.

Las complicaciones vinieron a raíz de terminar el primer prototipo, donde el motor DC iba encajado directamente a la rueda y, obviamente, el coche no se movió. Pensando que se debía a que los motores no tenían la suficiente capacidad para poder mover el coche, fueron cambiados por otros más potentes de un antiguo coche radio control. Ante mi ignorancia en la materia mecánica, encajé directamente los nuevos motores a las ruedas y el coche siguió sin moverse.

Fue entonces donde reparé en la principal característica de los motores DC, que tienen gran velocidad pero poco par. Por lo que se requería de un sistema de engranajes para cambiar esas propiedades y conseguir que el coche funcionase. De esta forma comencé a estudiar el sistema de engranajes reductor para poder diseñar uno. La ventaja con la que contaba es que los nuevos motores llevaban pegados el primer engranaje del sistema, por lo que la medida de los dientes debería de ser iguales para todos.

A continuación se detalla el desarrollo de las operaciones necesarias para el diseño del reductor:

Parámetros

- Diámetro primitivo del motor = D_p
- Número de engranajes = Z
- Módulo = $M = \frac{D_p}{Z}$
- Paso circular = $P_c = \pi \cdot M$
- Diámetro exterior = $D_e = D_p + 2 \cdot M$
- Grueso del diente = $S = \frac{P_c \cdot 19}{40}$
- Hueco del diente = $W = \frac{P_c \cdot 21}{40}$
- Diámetro interior = $D_i = D_p - 2 \cdot 1,25 \cdot M$
- Cabeza diente = M
- Altura diente = $2,25 \cdot M$

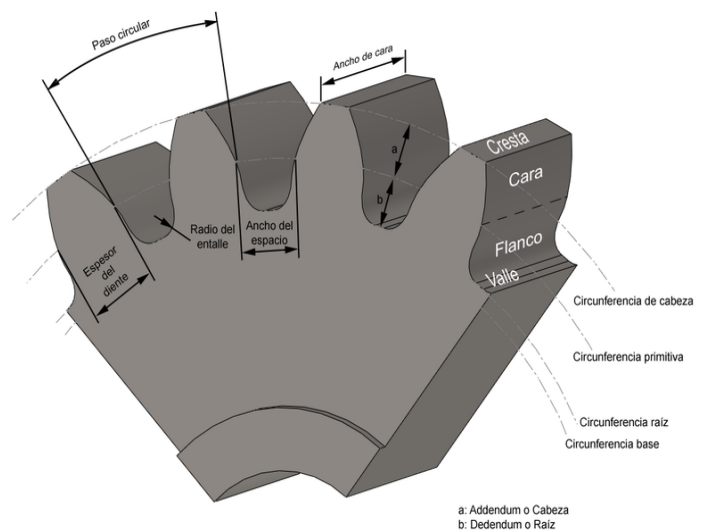


Figura 44. Partes de un engranaje

Los datos iniciales con los que contaba gracias a tener el primer engranaje eran los siguientes:

Engranaje 1

$D_p = 6\text{mm}$ y $Z = 8$. A partir de ellos sacamos el resto de variables:

$$M = 0,75 \quad P_c = \pi \cdot M = 2,356 \text{ mm} \quad D_e = D_p + 2 \cdot M = 6 + 2 \cdot 0,75 = 7,5 \text{ mm}$$

$$S = \frac{P_c \cdot 19}{40} = \frac{2,356 \cdot 19}{40} = 1,11 \text{ mm} \quad W = \frac{P_c \cdot 21}{40} = \frac{2,356 \cdot 21}{40} = 1,23 \text{ mm}$$

$$D_i = D_p - 2 \cdot 1,25 \cdot M = 6 - 2 \cdot 1,25 \cdot 0,75 = 4,125 \text{ mm}$$

$$\text{Altura diente} = 1,6875 \text{ mm}$$

Una condición para que se transmita el movimiento entre engranajes es que todos han de tener el mismo módulo, por lo que $M_1=M_2=M_3$. Con ello conseguimos que todos los datos relacionados con los dientes del engranaje (M, P_c , S, W y altura del diente) sean igual para todos.

Como del tamaño no había ningún dato, observé la estructura de las reductoras e impuse la medida del diámetro interior del resto de engranajes según el tamaño que me convenía para el coche:

Engranaje 2

$$D_i = 20 \text{ mm (dato impuesto)}$$

$$D_i = D_p - 2 \cdot 1,25 \cdot M \rightarrow D_p = D_i + 2 \cdot 1,25 \cdot M = 21,875 \text{ mm}$$

$M = \frac{D_p}{Z} \rightarrow Z = \frac{D_p}{M} = \frac{21,875}{0,75} = 29,16$ como el número de engranajes no puede ser decimal, aproximo por arriba a $Z=30$. Pero si añado un diente más, manteniendo los diámetros, el hueco entre dientes no será el mismo que en el engranaje 1, por lo que no habrá transmisión. Por lo que vuelvo a calcular los diámetros imponiendo ahora el dato de $Z=30$.

$$D_p = M \cdot Z = 22,5 \text{ mm} \rightarrow D_i = D_p - 2 \cdot 1,25 \cdot M = 20,625 \text{ mm}$$

Engranaje 3

$$D_i = 10 \text{ mm (dato impuesto)}$$

$$D_i = D_p - 2 \cdot 1,25 \cdot M \rightarrow D_p = D_i + 2 \cdot 1,25 \cdot M = 11,875 \text{ mm}$$

$$M = \frac{D_p}{Z} \rightarrow Z = \frac{D_p}{M} = \frac{11,875}{0,75} = 15,83$$

Por lo que aproximo $Z=16$

$$D_p = M \cdot Z = 12 \text{ mm} \rightarrow D_i = D_p - 2 \cdot 1,25 \cdot M = 10,125 \text{ mm}$$

Engranaje 4

$D_i = 28 \text{ mm}$ (dato impuesto)

$$D_i = D_p - 2 \cdot 1,25 \cdot M \rightarrow D_p = D_i + 2 \cdot 1,25 \cdot M = 29,875 \text{ mm}$$

$$M = \frac{D_p}{Z} \rightarrow Z = \frac{D_p}{M} = \frac{29,875}{0,75} = 39,83$$

Por lo que aproximo $Z=40$

$$D_p = M \cdot Z = 30 \text{ mm} \rightarrow D_i = D_p - 2 \cdot 1,25 \cdot M = 28,125 \text{ mm}$$

Y por fin tenemos nuestro sistema reductor:

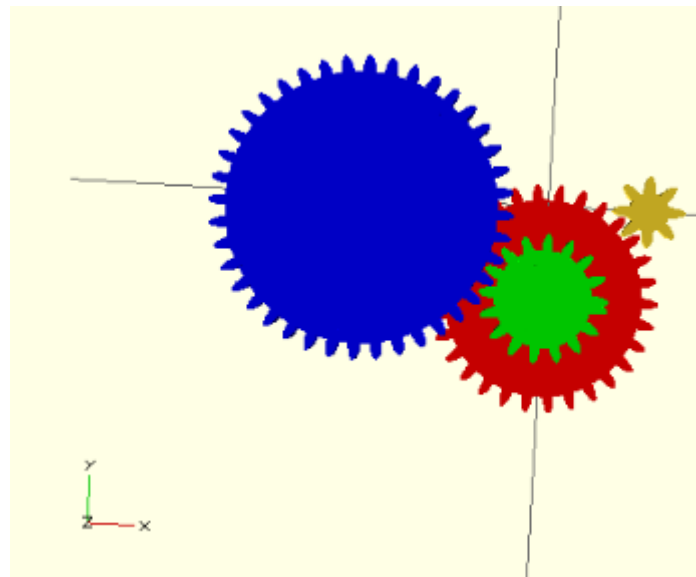


Figura 45. Sistema Reductor obtenido

Ahora comprobaremos las características que tiene este reductor:

$$V_i = r_i \cdot \omega_i = \frac{d_i}{2} \cdot \omega_i$$

El punto donde entran en contacto dos engranajes, se da que ambos tienen la misma velocidad lineal en ese punto:

$$V_1 = V_2 \rightarrow d_1 \cdot \omega_1 = d_2 \cdot \omega_2$$

Siendo d_i el diámetro interno:

$$4,125 \cdot \omega_1 = 20,625 \cdot \omega_2 \rightarrow \omega_2 = 0,2 \cdot \omega_1 = \omega_3$$

w_2 y w_3 son iguales debido a que los dos engranajes están pegados y situados en el mismo eje de rotación.

$$d_3 \cdot w_3 = d_4 \cdot w_4$$

$$10,125 \cdot 0,2 \cdot w_1 = 28,125 \cdot w_4 \rightarrow w_4 = 0,072 \cdot w_1$$

Esto significa que, suponiendo máxima velocidad del motor $w_1 = 20040 \text{ rpm}$ (véase Anexo A.1), nuestra rueda girará $w_4 = 1442,88 \text{ rpm}$ o lo que es lo mismo $24,05 \text{ vueltas/s}$. Ahora, suponiendo que la rueda no resbala, en cada vuelta ésta hace $P = 2 \cdot \pi \cdot r = 2 \cdot \pi \cdot 3,5 \text{ cm} = 22 \text{ cm/vuelta}$, o cambiando de unidades $529,05 \text{ cm/s} = 5,29 \text{ m/s} = 19,04 \text{ Km/s}$

Aunque hemos reducido en gran medida la velocidad de los motores, seguimos obteniendo un alto valor para este coche. En teoría, nunca se alcanzará ese valor máximo de velocidad puesto que los nuevos motores tienen una tensión nominal de $9,6 \text{ V}$, pero los estamos alimentando con una batería de $7,4 \text{ V}$ de máxima. No se ha modificado la batería para cumplir con la tensión de los motores porque tras las pruebas realizadas, el funcionamiento es correcto y con menor velocidad tienen mayor control de dirección.

4.3.6 Circuito de control de motores

Para poder mover los motores en ambos sentidos, se ha implementado un circuito extra que integra el componente L293B. Para ello, se ha seguido el esquema mostrado en las hojas de características, quedando nuestro circuito de la siguiente manera:

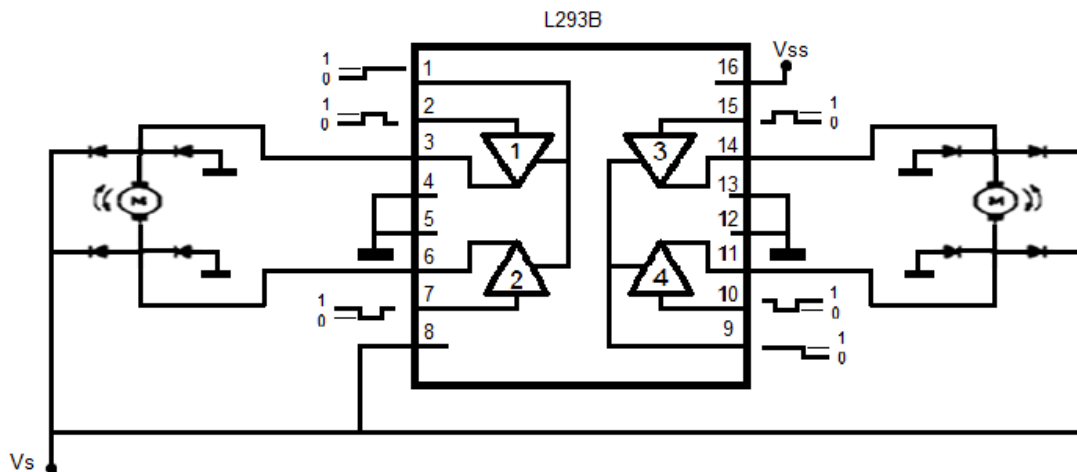


Figura 46. Esquema del circuito

Donde los pines 2, 7, 10 y 15 son las señales que van directo a los pines de programación de arduino (pines digitales), y 1 y 9 son los chips enable que permiten que cada motor funcione o no. Como siempre vamos a usar ambos motores, hemos unido esos dos pines con el de V_{ss} (alimentación lógica o pin PWM de Arduino que dará la información PWM a

los motores), para que siempre que el circuito reciba la información requerida, funcionen. Y por último, Vs es la alimentación, por lo que irá directamente conectado a la batería.

Para poder realizar este circuito, además hemos requerido de diodos 1N4001 (véase Anexo A.2), para poder hacer el esquema, y de una placa de puntos para poder soldar y unir todos los componentes. De forma que el resultado ha sido el siguiente:

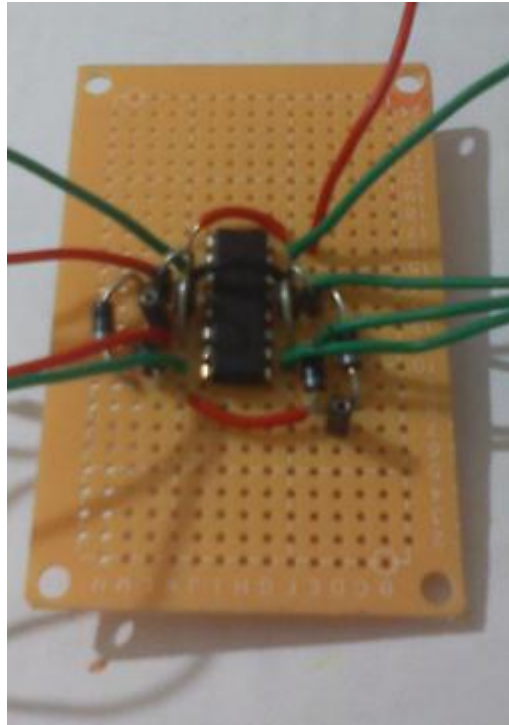


Figura 47. Resultado del circuito

Gracias al circuito integrado nos hemos ahorrado el implementar un gran número de componentes, simplificando el circuito y quedando el resultado disponible en un tamaño reducido y manejable para posicionar donde nos convenga. Además, nos evitamos errores de conexionado (puesto que soldar una placa sin la capa que nos indica donde va cada componente, es más fácil equivocarse de conexión y más cuando el número de componentes es elevado).

4.4 Fabricación y Montaje

Para la fabricación de nuestro modelo primero hemos de tener en cuenta los diseños realizados de las piezas mencionadas en el apartado anterior. Todas ellas fueron realizadas y parametrizadas mediante lenguaje OOML2 o el programa OpenScad, de forma que obtenemos tres archivos: .cpp, .scad y .stl. El primero de ellos es para las que fueron hechas en lenguaje OOML2, de forma que si se requiere alguna modificación en ellas se debe realizar en este archivo. En el caso de que fueran hechas solo con OpenScad, los cambios se realizarían en el .scad. Finalmente el último archivo es el que usaremos para poder imprimir la pieza en la impresora 3D.

Una vez tenemos todas las piezas como deseamos, para obtenerlas tan sólo debemos imprimirlas en 3D. Para comprobar que se tienen todas las piezas se facilita la siguiente lista:

- Parte delantera:
 - a. Pieza delantera central
 - b. Pieza delantera lateral x2
 - c. Rueda con rodamientos x2
 - d. Parte superior suspensión x2
 - e. Parte inferior suspensión x2
 - f. Pieza de dirección derecha
 - g. Pieza de dirección izquierda
 - h. Pieza unión servo con dirección
 - i. Soporte Servo
 - j. Pieza de unión entre el soporte servo y la pieza delantera x2
- Parte trasera:
 - a. Engranaje número 1 x2
 - b. Engranaje número 2y3 x2
 - c. Engranaje número 4 x2
 - d. Soporte de engranajes trasero del lado izquierdo
 - e. Soporte de engranajes trasero del lado derecho
 - f. Soporte de engranajes delantero del lado izquierdo
 - g. Soporte de engranajes delantero del lado derecho
 - h. Soporte Motor x2
 - i. Ruedas con suspensión x2
 - j. Pieza Soporte Circuito ("Maletero")
 - k. Soportes para la placa arduino x8

Por lo que en total contamos con 36 piezas imprimibles. A continuación se detallará el montaje completo de cada una de ellas con los diferentes elementos que componen el total del coche.

4.4.1 Montaje Parte Delantera

Comenzamos el montaje de la parte delantera por las suspensiones. En primer lugar introducimos la parte inferior de la suspensión dentro de la parte superior. Luego introducimos el muelle por ambas partes y lo centramos. Una vez tenemos el muelle bien colocado, cogemos una goma elástica fina (lo suficiente para que quepa en el hueco que tiene la parte inferior de la suspensión) y la cortamos por la mitad. Uno de los extremos de la goma lo introducimos con cuidado a través del agujero de la parte inferior de la suspensión y hacemos un nudo alrededor de ella. Se deberá repetir el proceso para la suspensión del otro lado.

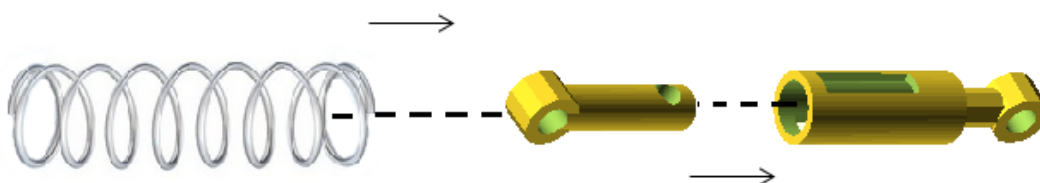


Figura 48. Montaje Suspensión I parte



Figura 49. Montaje Suspensión II parte

Para terminar la colocación de la suspensión, cogeremos las piezas delantera central y lateral. En primer lugar deberemos unir éstas entre sí mediante un tornillo de dos milímetros de diámetro y 2,5cm de largo, con rosca solo los últimos 0,5 cm si se quiere que el deslizamiento entre piezas sea mejor. En caso de no disponer de esos tornillos, bastaría con modificar las piezas para el tornillo disponible.

Una vez unidas, unimos la parte superior de la suspensión con la pieza delantera central mediante tornillos de 16mm de métrica 3 (m3) y tuercas, y la parte inferior de la suspensión con la pieza delantera lateral también mediante tornillos de 16 mm y tuercas m3.

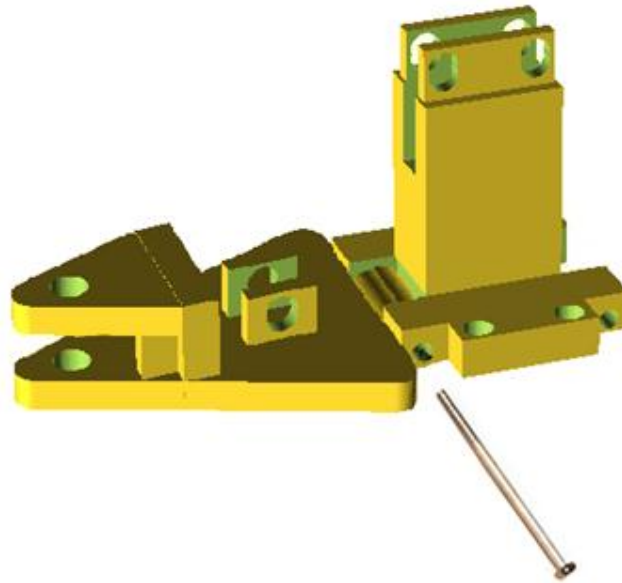


Figura 50. Montaje piezas delanteras

Para finalizar el conjunto delantero hemos de unir la pieza de dirección con la pieza lateral y a su vez, ésta con el rodamiento y la rueda. Es aconsejable unir primero la dirección con la del rodamiento y la rueda, puesto que luego unir la dirección a la pieza lateral es mucho más fácil, dado que es solo unir con un tornillo m4. La dirección debe encajar perfectamente en el rodamiento y éste en la rueda para evitar holguras y que el conjunto tenga un correcto funcionamiento. Cuando se una la dirección con la pieza lateral se debe tener en cuenta que el cuadrado debe estar mirando al interior del conjunto, para el posterior montaje con el servo y porque esa pieza no es igual para el lado izquierdo que para el derecho.

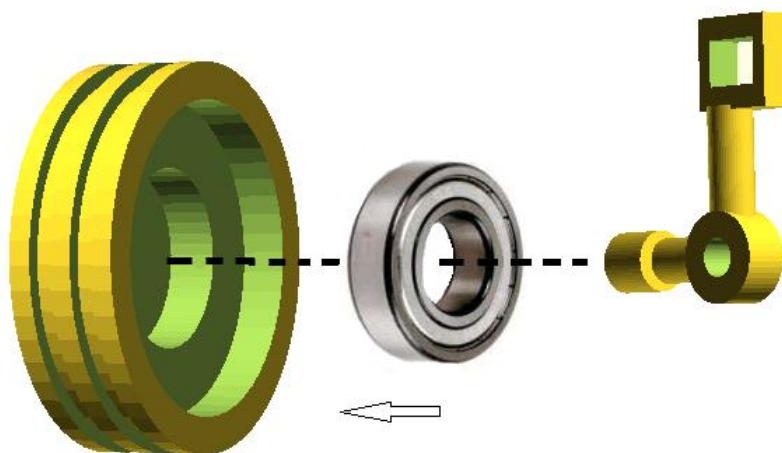


Figura 51. Montaje pieza dirección con rodamiento y rueda

Los huecos que se observan en la rueda son para añadir juntas tóricas de 4,5 cm de diámetro, para conseguir que la rueda agarre mejor y no resbale.

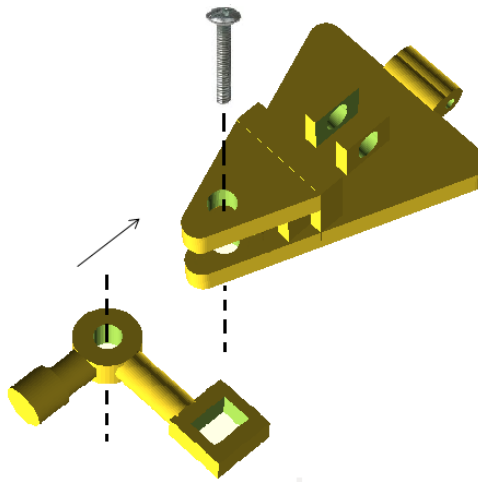


Figura 52. Montaje pieza lateral con pieza dirección

El siguiente paso es montar la pieza que nos ayudará a dar dirección al conjunto, que va unida al servo. Basta con usar los tornillos que trae el servo consigo y colocarla donde se ponen los accesorios con los que viene. Hemos de colocarla con el servo calibrado para que el giro luego nos sirva para la dirección. Para ello, lo mejor es tumbar el servo y cuando se encuentre perpendicular al suelo (da igual a la izquierda o a la derecha), calibrarlo para que esa medida sea 0 ó 180°.

Una vez calibrado, introduciremos el servo en su soporte sujetándolo mediante 4 tornillos m4.

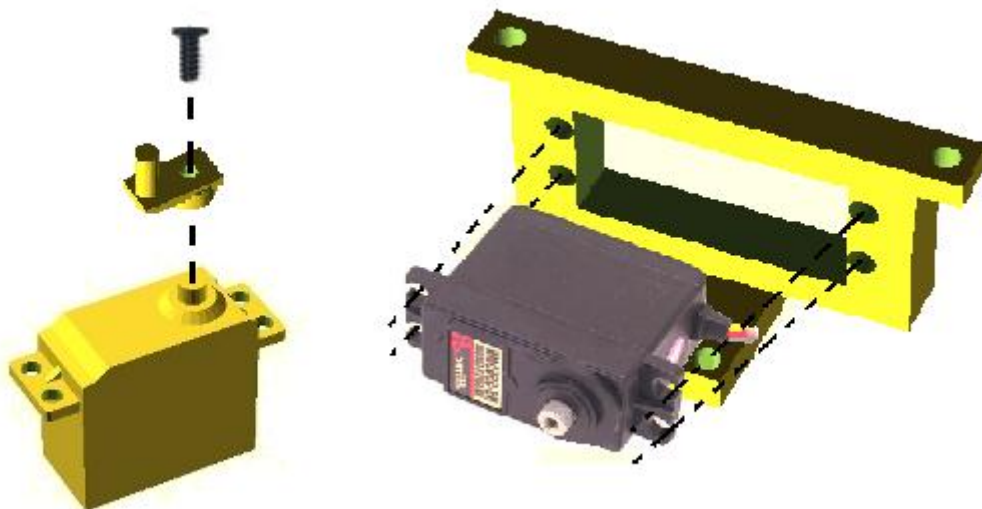


Figura 53. Montaje del servo

Una vez tenemos el servo bien posicionado, pasamos a unir la parte delantera central con el soporte del servo, mediante dos piezas de unión (una por la zona de arriba y otra por la de abajo, para reforzar la unión) con tornillos de 16 mm m4.

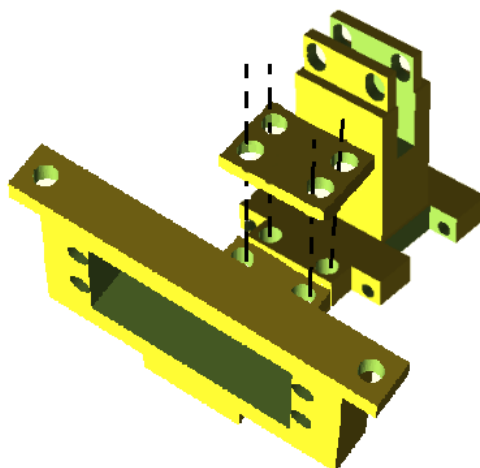


Figura 54. Unión de la parte delantera con el soporte del servo

Para terminar, cogemos alambre y lo entrelazamos (para evitar que se flexione) y pasamos un extremo por el interior del cuadrado de la pieza de dirección y el otro por la pieza que está unida al servo y que se moverá. Habremos de hacerlo con cuidado para que las ruedas estén rectas y ajustarlo lo máximo posible para poder realizar el giro correctamente.



Figura 55. Resultado final de la parte delantera

De esta forma finalizamos la parte delantera. Si alguna parte queda muy holgada o inestable, es aconsejable usar pegamento de silicona (siempre que se pueda).

4.4.2 Montaje de la parte trasera

El montaje de la parte trasera es más fácil, aunque hay más lío de cableado. Para empezar, montaremos el sistema de engranajes reductor. Para ello encajamos el engranaje 1 en el motor y lo pegamos para evitar que la pieza resbale y no transmita el movimiento. Luego cogemos uno de los soportes de los engranajes trasero (¡cuidado! No son iguales, dependen del lado en el que lo coloques) y superponemos los engranajes 2 y 4 (el tercero está contenido en el número 2) sobre él. Una vez colocados, ponemos el soporte delantero para que no se muevan de su posición los engranajes y unimos al trasero mediante 4 tornillos m3 (3 tornillos

de 30mm y uno de 20mm, el de la esquina superior al agujero del soporte trasero por donde se introducirá el engranaje unido al motor. Esto es debido a que esa esquina estará unida al soporte motor y con tornillos más largos esa unión entraría en conflicto). Se ha de tener mucho cuidado de no apretar los tornillos demasiado porque si no, los engranajes no podrán realizar el giro.

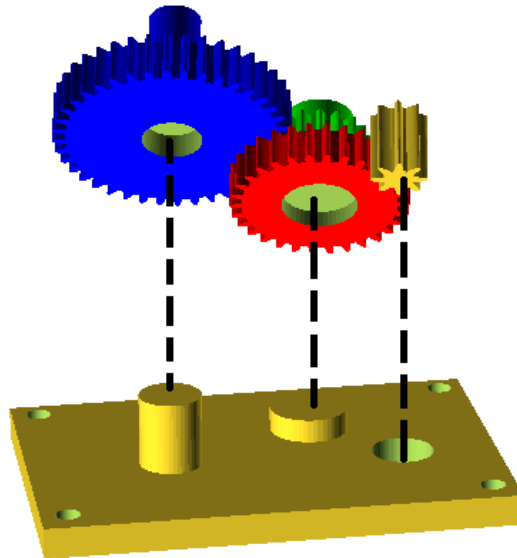


Figura 56. Colocación engranajes en soporte trasero derecho

Es aconsejable lubricar o engrasar los engranajes para que tengan un movimiento menos agresivo para las piezas.

El siguiente paso es encajar el motor en su soporte. Debería de entrar de forma sobrada y para evitar movimiento se debería pegar con un poco de silicona. El soporte es más corto que el propio motor para dejar los agujeros de ventilación de que dispone, libres y pueda ventilarse.

Una vez tenemos el motor en su soporte y los sistemas de engranajes, unimos mediante pegamento de silicona ambos. Las dos piezas han de quedar a la misma altura, para que cuando se una con la base principal no se mueva innecesariamente.



Figura 57. Colocación de engranajes y motor

El último paso para terminar la parte trasera es la colocación de la rueda con suspensión. Antes de colocarla, podemos añadirle goma para que no deslice con el suelo. En mi caso, he recortado goma de 5mm de ancho en varios trozos de aproximadamente 10 mm de largo, y lo he colocado a intervalos regulares a lo largo de la rueda.

Para colocar la rueda correctamente, se ha de dejar un espacio con respecto al soporte de engranajes porque sobresalen las cabezas de los tornillos, por lo que es importante que la rueda no roce con ellos.

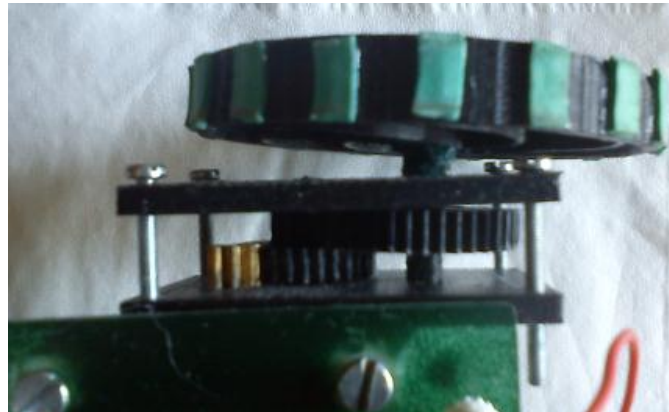


Figura 58. Montaje de engranajes y rueda trasera

4.4.3 Montaje con la base

Ahora que tenemos las partes delantera y la trasera montadas, ahora debemos unirlo con la base de metacrilato. Para ello, en primer lugar hemos de cortar el metacrilato con la medida deseada, en este caso de 20x13 cm.

Hemos de añadir también 6 agujeros de métrica m4 para poder unir el soporte del servo y los soportes del motor. Es aconsejable no hacerlos muy al borde de la placa por si ésta partiera. La configuración que he realizado se puede ver en la siguiente figura:

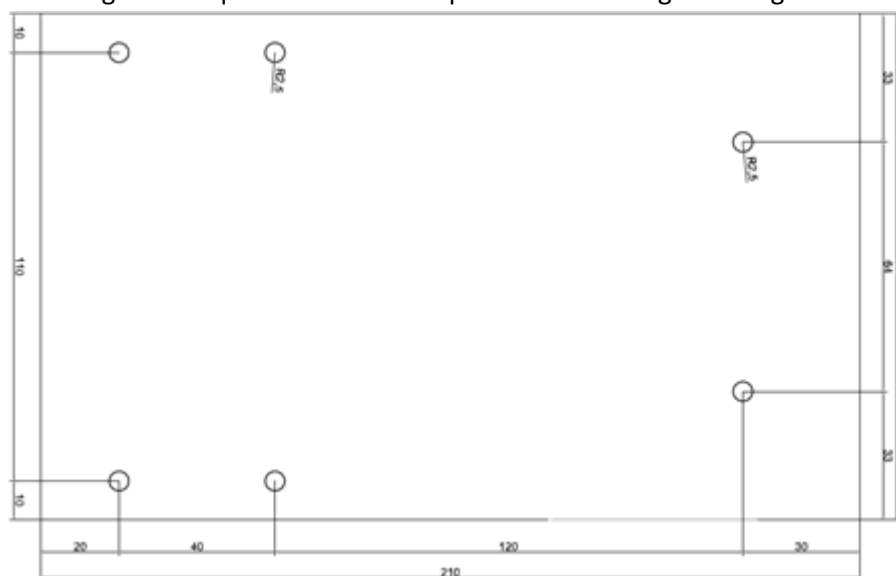


Figura 59. Configuración de la placa de metacrilato

Una vez tenemos realizados los agujeros, solo debemos unir con tornillos m4 cada una de las partes mencionadas anteriormente. Si alguna de las partes queda suelta, conviene reforzar con pegamento de silicona. Además, si los apoyos de la placa están tan separados, como es el caso, se debe fortalecer poniendo otra placa de metacrilato debajo. Con que el tamaño sea de 16 x 4 cm es suficiente para situarlo en la parte central tras el soporte del servo.

Por último, solo debemos pegar en la parte superior de la placa los soportes para la placa arduino y el circuito con el puente H, donde mejor nos parezca. En este caso, se ha situado en la zona trasera para dejar la parte central para posibles cargas que pueda llevar. Cuando sepamos donde lo vamos a situar, lo mejor es unir mediante tornillos los soportes con las diferentes placas, para tener la posición exacta de los soportes respecto de la placa y posteriormente pegarlos al metacrilato.

El soporte del puente H no necesita soportes extra, puesto que viene con apoyos, pero se los he añadido para ganar altura y poder introducir en el hueco restante la batería. De esta forma finalizamos el montaje de nuestro coche:



Figura 60. Montaje completo del coche

4.5 Implementación

Ya terminado el montaje mecánico del coche, pasamos al control y las comunicaciones para poder empezar a probar los movimientos del coche. Así podemos empezar a hacer las primeras pruebas para modificar y corregir lo que sea necesario para que funcione correctamente.

4.5.1 Control mediante GamePad

Para poder realizar el control de nuestro vehículo mediante un mando externo al ordenador, hemos de realizar un programa en lenguaje C++ que sea capaz de leer los datos obtenidos por el mando, procesarlos y enviarlos a la placa arduino, para que ésta a su vez procese los datos y se activen los elementos necesarios. Dicho programa es una adaptación del creado por Alberto Valero y Julián Marín. En este proyecto se ha usado el entorno *Qt Creator* como compilador de C++, puesto que ordena las clases de una forma clara y entendible visualmente. El programa realizado está formado por tres clases:

- **Joystick:** Es la clase encargada de leer los datos que envía el mando. Gracias a ella, podemos seleccionar los botones y joysticks que necesitamos usar. Calibrará todos los botones del mando y, si lo necesitamos, los normalizará entre los valores de -1 a 1, para poder luego operar más fácilmente con ellos. Se deberá indicar el puerto del ordenador por donde se ha conectado el mando para realizar la comunicación con él.
- **ArduserialStream:** Es la encargada de gestionar el flujo de datos por el puerto serie entre el ordenador y la placa arduino. Se deberá seleccionar también el puerto al que está conectada la placa arduino, ya sea por usb o por Bluetooth.
- **Gamepad2Arduino:** Esta clase es la combinación de las otras dos, es decir, hereda de las otras dos. Con ella podemos realizar lo de las dos clases anteriores más fácilmente y hacer las operaciones pertinentes que necesitamos que reciba arduino. En nuestro caso se han seleccionado los dos gatillos analógicos (nombrados en el mando por RT y LT), el joystick derecho y los botones A y B. Los tres primeros han sido calibrados para dar los valores requeridos por arduino para poder mover la dirección y los motores, mientras que los botones A y B dan los valores 0 ó 1.

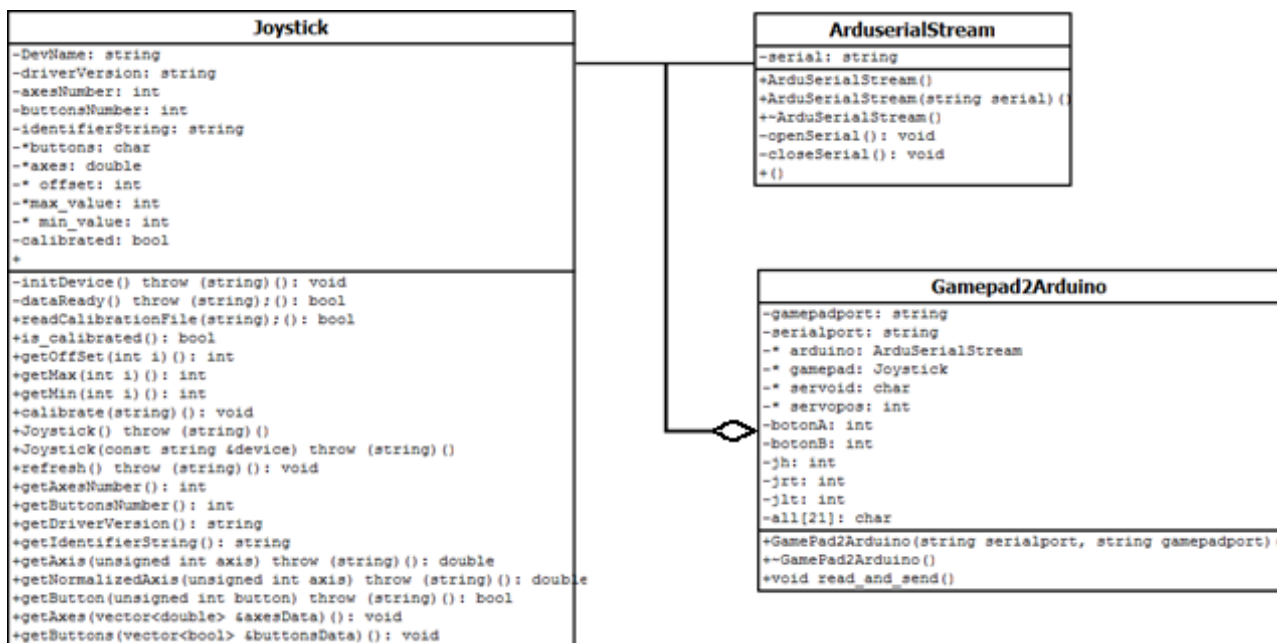


Figura 61. Diagrama de clases

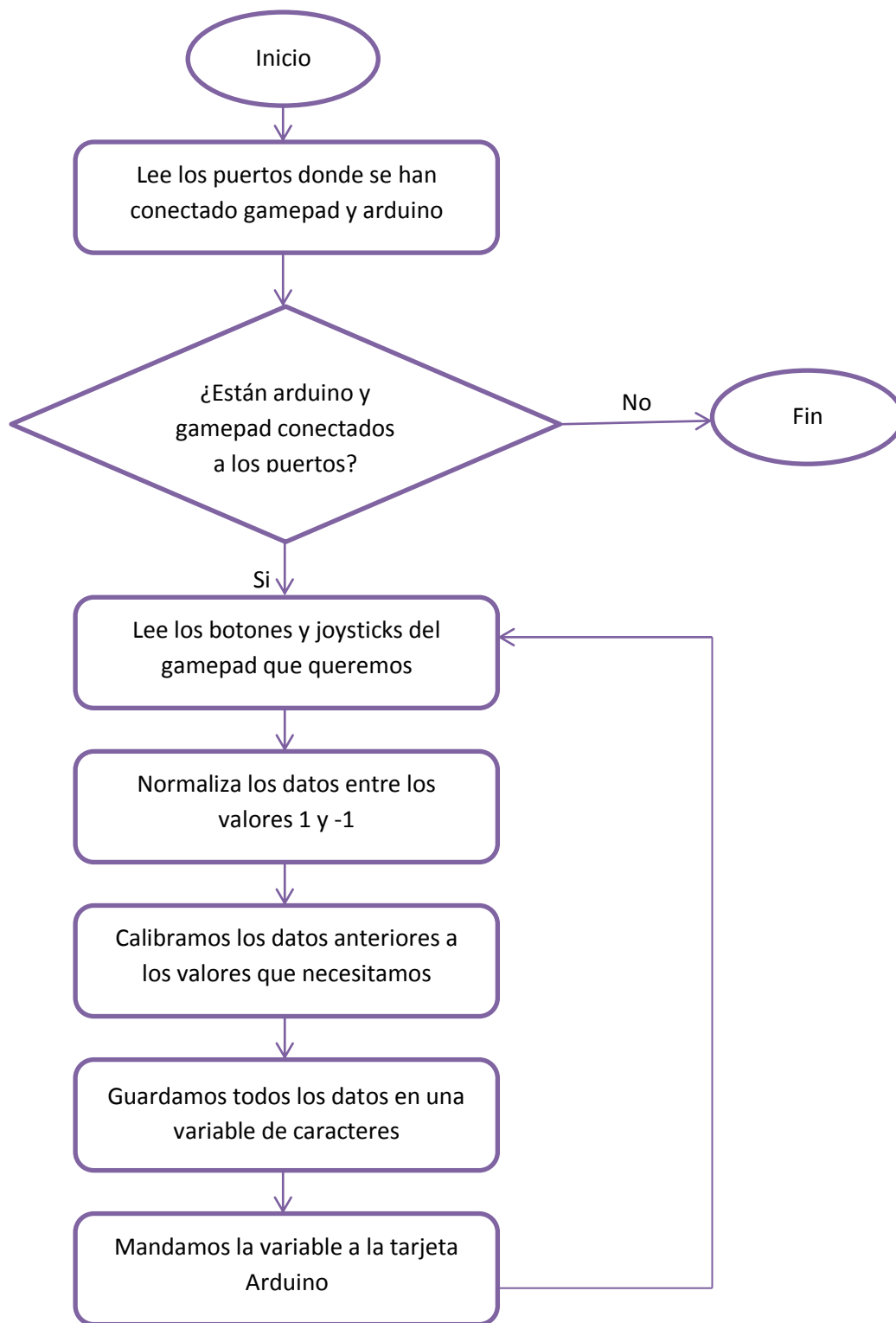


Figura 62. Flujograma del programa de C++

Con ello conseguimos leer los datos, procesarlos y mandarlos a arduino, ¿pero qué hace exactamente arduino? Pues es muy sencillo: la placa arduino tiene a su vez otro programa que le hemos cargado previamente que se encarga de dar sentido a los datos que recibe, es decir, él solo recibe una cadena de caracteres con todos los datos que hemos requerido del mando. Éste programa separa los datos de cada botón/joystick y aquí es donde

nosotros le programamos lo que queremos que haga el vehículo. Dicho programa es una modificación del creado por Ana De Prado.

En este caso le hemos indicado al programa que haga las siguientes acciones: los gatillos analógicos están calibrados de 0 a 254, dado que indicarán el valor de PWM que queremos que tengan los motores, siendo cero un ciclo de trabajo nulo y 254 casi un 100% del ciclo de trabajo realizado (Arduino permite 8 bits = 256 valores de PWM, no se ha añadido el valor 255 para evitar tratar con decimales en la calibración). Si pulsamos el gatillo derecho el coche irá hacia delante y con el gatillo izquierdo hacia atrás. El joystick derecho está configurado para que tenga los valores de 0 a 180, para indicar al servo la dirección que ha de tomar. Cuando el botón B es pulsado se enciende un LED y se mantiene hasta que volvamos a pulsarlo. Y finalmente, el botón A es el freno de emergencia. Cuando es pulsado el ciclo de trabajo es directamente cero, haciendo que los motores se paren al instante.



Figura 63. Botones usados del gamepad

4.5.2 Comunicación inalámbrica

Nuestro vehículo ya está listo y programado para funcionar, pero está limitado por el largo del cable del puerto serie. Por lo que el siguiente paso es conseguir que arduino se comunique con el ordenador mediante el módulo de Bluetooth. Para ello, conectamos primero dicho módulo a la placa de la siguiente forma:

- Pin Rx del módulo al pin Tx de la placa
- Pin Tx del módulo al pin Rx de la placa
- Pines GND y VCC de la placa a los pines GND y 5V de la placa

Una vez enchufado correctamente, hemos de buscarlo por el ordenador para que se conecten entre sí. En caso de usar sistema Linux, es aconsejable usar el programa de Bluetooth Blueman, puesto que los otros no localizan correctamente este módulo. Una vez lo veamos en el ordenador deberemos emparejarlo mediante la clave 1234. Nos dirá que ha habido un error pero se habrá emparejado correctamente y ahora solo debemos dar a configurar como puerto serie. Si se ha conectado aparecerá una ventana en el ordenador diciendo que se ha conectado el puerto serie dev/rfcomm0 (es el nombre del puerto que deberemos poner en el programa

de C++ mencionado anteriormente). En caso de que no se conecte, se deberá borrar la conexión y volverlo a intentar, dado que éste módulo falla mucho en este aspecto.

Hemos de tener cuidado con el programa de arduino, puesto que dependiendo de la placa que utilicemos deberemos modificar algunos datos con respecto al programa normal de comunicación mediante usb. En este caso, la placa Leonardo necesitó cambiar los comandos **Serial** del programa (los que nos ayudan a configurar el puerto por el que arduino recibe y manda datos) por **Serial1**. Esto fue debido a que Leonardo está configurado de forma que la comunicación USB se reciba mediante el primer comando y el segundo es para la comunicación TTL (transistor-transistor logic) la cual usa el módulo Bluetooth.

Cuando se ha conectado todo correctamente, el funcionamiento del vehículo ha de ser igual que con el cable usb, puesto que las acciones que realiza no han sido modificadas y así contamos con el rango de 10 m de distancia para manejarlo sin cables.

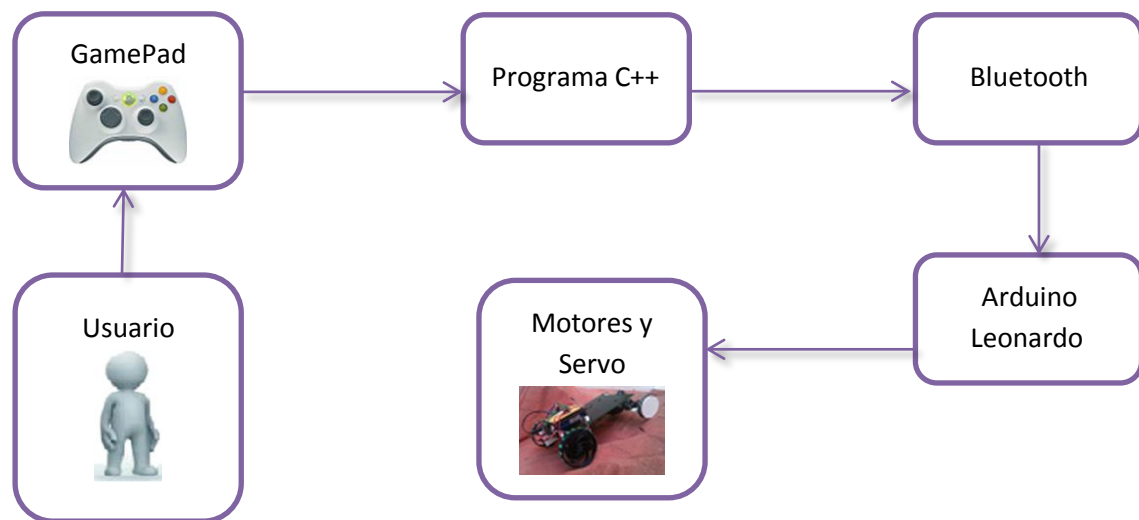


Figura 64. Diagrama de bloques del funcionamiento del sistema

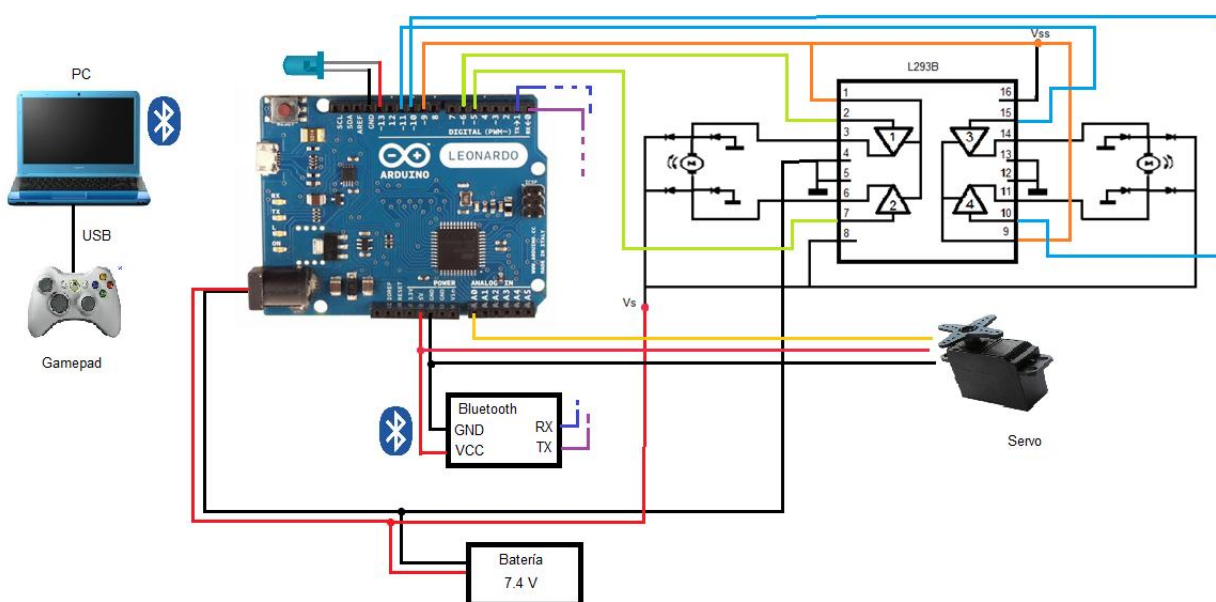


Figura 65. Conexión total del sistema

4.5.3 Pruebas y modificaciones

Esta es la parte más importante de todo proyecto, puesto que se comprueba el funcionamiento total y es el momento para reflexionar y ver qué modificaciones podríamos hacer.

Los cambios en este proyecto han sido muy variados y visibles hasta llegar al prototipo final, y aun así, seguro que podríamos seguir mejorándolo. Algunos de los cambios realizados han sido los siguientes:

- Las piezas central y lateral de la parte delantera han sido creadas y modificadas cinco veces hasta llegar al aspecto actual. Esto fue debido a que cada vez que acababa de imprimir ese conjunto le encontraba un defecto que durante el diseño no había visto, o mejoraba las partes para que quedase más ajustado al grupo.
- Las ruedas delanteras, en el primer prototipo, no tenían rodamientos, sino que iban unidas mediante tornillos de 2mm de diámetro con rosca solo al final. El problema resultó ser que se desenroscaban y el tornillo tenía demasiada holgura con respecto a la rueda de dirección, provocando un movimiento exagerado de la rueda.
- La dirección ha sido probada a realizar mediante piezas imprimibles, bridas y finalmente alambre.
- Las ruedas traseras iban unidas directamente a los motores, por lo que se cambió por otros motores más potentes pero el coche no andaba, a lo que finalmente se hizo un sistema de engranajes para cada una. El problema es que con los engranajes perdió altura con respecto al suelo, por lo que ahora es más vulnerable a los obstáculos altos.
- El vehículo iba a usar inicialmente la placa Skymega (creada por Juan González), pero durante el proceso se estropeó, teniendo que adquirir otra placa parecida, la cual fue Arduino Leonardo. La ventaja de la primera con respecto a otras placas arduino es que es más pequeña, los pines para los servos están colocados de forma que no tienes que estar sacando cables extra y es más barata. La desventaja con respecto a Leonardo es que el microprocesador es más antiguo.
- Se pensó en hacer un control diferencial del vehículo para mejorar la maniobrabilidad, pero tras estudiar la cinemática y el que las ruedas delanteras fueran independientes de las traseras, era imposible.
- Algunas piezas tuvieron que ser diseñadas con partes de más para que fueran imprimibles. Otras, como la pieza de dirección, no son iguales para un lado que para el otro por el mismo motivo.
- En principio las ruedas traseras iban a ser iguales que las delanteras pero de mayor tamaño, pero al explorar por www.thigiverse.com y encontrar las ruedas con suspensión incorporada, se decidió por la segunda idea.
- Originariamente se pensaba realizar el proyecto con dos servos trucados, es decir, que girasen 360°, pero todos los robots imprimibles creados hasta el momento usaban eso y no provocaba ninguna novedad en este campo.

- Cuando se usaban los primeros motores, se pensó en alimentarlos a 6 V. Como la batería era de 7.4 V se adquirió un adaptador llamado UBEC, que pasa dicho voltaje a 6 ó 5 V, según se requiera. Pero al cambiar los motores, se decidió alimentarlos directamente a 7,4 V, puesto que su máximo eran 9,6 V, por lo que el UBEC fue quitado del sistema.



Figura 66. Alguna de las modificaciones realizadas

Además, se tuvo que tener en cuenta que los motores a partir del valor 100 para el PWM del programa arduino (aproximadamente el 40-50% de ciclo de trabajo) no eran capaces de mover el coche, produciendo un sonido incómodo que nos indicaba que se encontraban en estado de bloqueo. No se le llegó a tomar importancia hasta que se comprobó que gastaban corriente innecesariamente para no provocar ningún movimiento en el coche. Por ello, se decidió que el programa de Arduino no tuviera en cuenta las señales recibidas por el mando hasta que no superasen un valor mayor o igual a 100 para el PWM. De esta forma nos evitamos el sonido de bloqueo y posibles fallos en el sistema debido a ello.

Y, por último, a parte de todas las modificaciones técnicas, se comprobó que el tamaño del vehículo fuera el idóneo para poder transportar un brazo robótico. Con las placas situadas en la parte trasera y los agujeros de los tornillos del servo lo suficientemente delante, no se estorbaba al brazo robótico. La única desventaja es que se ha de crear una pieza intermedia para poder encajar el robot en el coche, sin necesidad de pegarlo, para que pueda llevar otras cargas si fuera necesario o para usar el brazo robótico donde se requiera sin que esté obligado a permanecer en el coche.



Figura 67. Brazo Robótico y Vehículo

5. CONCLUSIONES Y TRABAJOS FUTUROS

Después de la realización completa del vehículo y de haber conseguido cumplir todos los objetivos estipulados, llego a las siguientes conclusiones:

Ha sido un trabajo arduo de aprendizaje durante meses, pero que ha merecido la pena porque ahora mis conocimientos sobre diferentes ámbitos han crecido y me ha permitido madurar como persona ante la aparición de diferentes problemas y la manera de resolverlos. Además esta experiencia me ha permitido conocer el creciente mundo de las impresoras 3D, con todo lo que se puede hacer con ellas, y la gran comunidad existente de OpenSource donde todo se comparte para mejorar las cosas de forma más fácil.

Bien es cierto que de la parte mecánica no tenía mucha idea y me ha resultado muy difícil realizar esa parte puesto que no tenía en cuenta todo lo necesario y tenía más fallos que aciertos. Pero gracias a ello he podido adentrarme un poco más en este mundo y en el de los coches de radio control, puesto que son temas que me gustaban y llamaban la atención, pero nunca había tenido la iniciativa u oportunidad de intentar entenderlos un poco más. Un intento de conocer más sobre los vehículos fue con la asignatura de *Aplicaciones de la Automática en Vehículos*, pero era tan amplio el temario y tan poco el tiempo de que se disponía que sólo daba tiempo a mencionar los sistemas de automática implantados en los vehículos, sin profundizar demasiado en ellos.

Con este trabajo también he podido desarrollar los conocimientos aprendidos de electrónica para poder entender los circuitos y poder implementarlo. Ha sido la parte que más me ha gustado, junto con las pruebas que iba realizando, puesto que es la parte que más conocía y más entretenida me parecía. Las pruebas también lo eran puesto que ver cómo todo el trabajo realizado merecía la pena cuando todo funcionaba o encontrar los fallos cuando no lo hacía era lo que más tiempo llevaba.

El proyecto nos enseña las partes básicas de las que consta un vehículo o coche de radio control en las materias de mecánica y electrónica. Aunque hay algunas partes, como el diferencial de los coches, que no han sido mencionadas puesto que al tener un motor por rueda no necesitaban ser implantados. La parte de programación nos enseña como el trabajo OpenSource es más llevadero que empezar de cero tú mismo, puesto que si algo funciona ¿por qué no utilizarlo? O incluso mejorarlo.

Además, con este proyecto se consigue que cualquier persona aficionada a este mundo pueda crearse uno propio imprimiendo éste o modificándolo a su gusto. Puede también añadir otras funciones para el coche con alguno de los muchos botones o joysticks que no han sido usados. En esta parte la imaginación es la que tiene prioridad.

En el siguiente enlace podemos ver una breve demostración del funcionamiento del vehículo: <http://www.youtube.com/watch?v=xFhlsxR6l8s&feature=youtu.be>

En el futuro se puede partir de este trabajo para desarrollar sistemas de control de bajo coste (sin necesidad de equipos de sensor o actuadores caros) para coches autónomos, puesto que los mencionados que participaron en el concurso Grand Challenge tenía cada uno de ellos un presupuesto de unos 500000\$ mínimo. Para ello podemos usar los sensores utilizados en el robot móvil llamado Protobot, que son sensores de ultrasonido y de infrarrojos de bajo coste. Otro tipo de sensor que puede ser utilizado son los de mapeado, es decir, sensores que van realizando un mapa de la zona por la que pasan. Éste sensor tiene un control más complicado que los anteriores, pero puede ser igual de válido.

Además se pueden estudiar otros controles, como puede ser el del aparcamiento. Y para conseguir una mejor simulación se deberían de modificar los engranajes para que el coche tuviera un único eje, a poder ser incluir el diferencial antes mencionado. Otra posible modificación, ya más complicada, sería conseguir que la tracción y dirección se encuentren juntas en el eje delantero.

Una parte que mejoraría mucho este diseño, sería hacer mediante programas de diseño una PCB del circuito que contiene el puente H, con los caminos hechos en cobre que facilitarían la soldadura a los principiantes y que contuviera una capa de grafiado para indicar donde va cada componente.

Y, por último, se podría adaptar el vehículo para participar en el Mars Challenge, una prueba que se hace en la Universidad Carlos III de Madrid para motivar a los alumnos a crear robots imprimibles, que consiste en moverse sobre una recreación de Marte y cumplir el objetivo de coger un objeto de una parte del territorio y llevarlo hasta otro punto.

6. BIBLIOGRAFÍA

Éstas son las páginas webs o documentos usados y consultados para el desarrollo del proyecto:

- [1] Manual del usuario del programa OpenScad (Febrero 2012)
http://en.wikibooks.org/wiki/OpenSCAD_User_Manual
- [2] Página principal del lenguaje OOML (the C++ Object Oriented Mechanics Library) (Marzo 2012) <http://iearobotics.com/oomlwiki/doku.php?id=start>
- [3] Thingiverse, Digital designs for real, physical objects (Abril 2012)
<http://www.thingiverse.com>
- [4] Página principal en inglés de Arduino (Julio 2012) <http://www.arduino.cc/>
- [5] Página principal de la wiki-robotics learobotics: Ingeniería, Electrónica y Aplicaciones fríkis (Agosto 2012)
http://www.iearobotics.com/wiki/index.php?title=P%C3%A1gina_Principal
- [6] Manual del usuario del programa ReplicatorG (Agosto 2012)
http://wiki.ultimaker.com/ReplicatorG_User_Manual
- [7] Wikipedia, la enciclopedia libre (Agosto 2012) www.wikipedia.org
- [8] Apuntes de la asignatura *Aplicaciones de la Automática en Vehículos* (Agosto 2012)
- [9] *Cinemática de robot móviles* por Omar Sánchez (Agosto 2012)
http://www.uhu.es/omar_sanchez/Omar's%20Resources/Cinematicavehiculos.pdf
- [10] Apuntes de la asignatura *Robótica Industrial* (Agosto 2012)
- [11] *Pose Estimation of Ackerman Steering Vehicles for Outdoors Autonomous Navigation* de Alejandro J. Weinstein y Kevin L. Moore (Agosto 2012)
<http://www.lcad.inf.ufes.br/wiki/images/b/b8/Ackerman-steering.pdf>

II. Planos

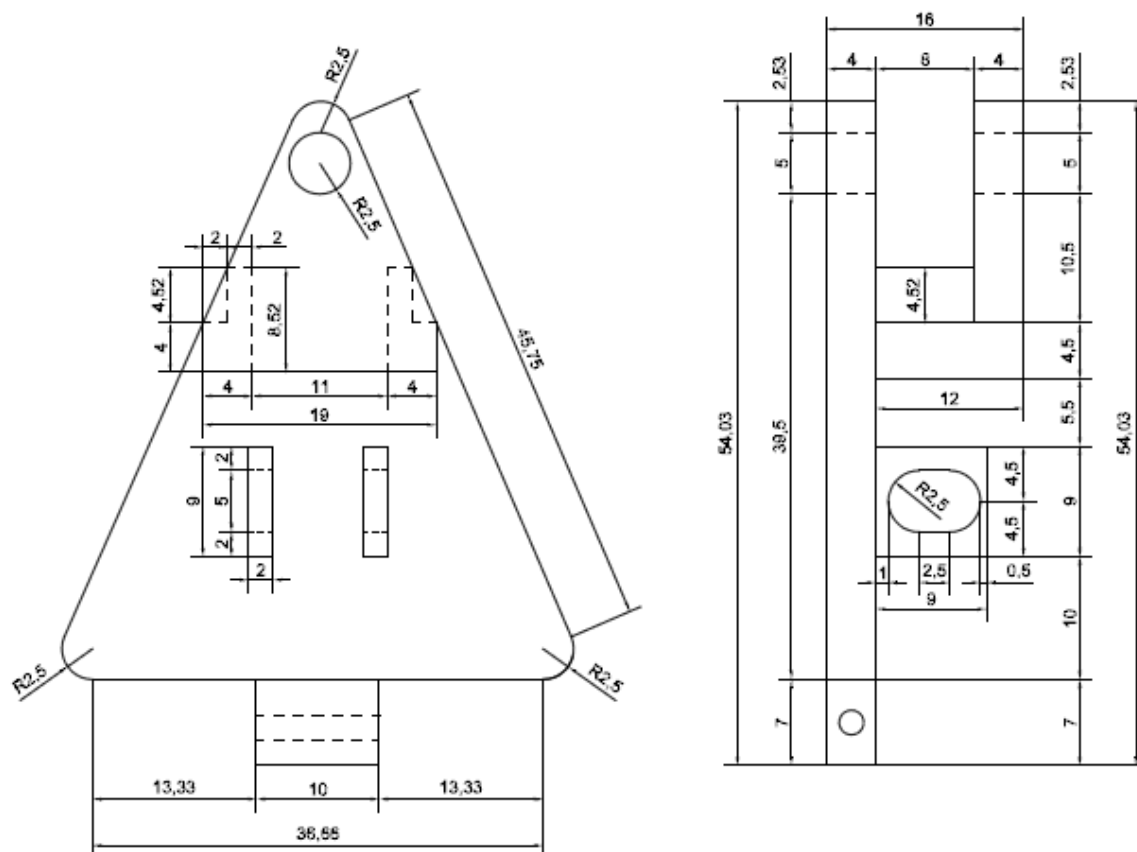
1. INTRODUCCIÓN


En esta parte se mostrarán los planos de las piezas diseñadas e impresas en las impresoras 3D de este proyecto. Los planos han sido realizados en el programa AutoCAD 2013 y han sido acotados en milímetros.

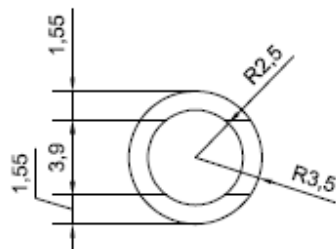
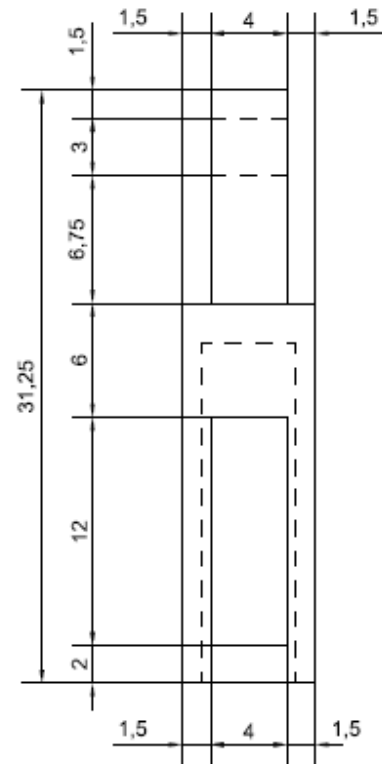
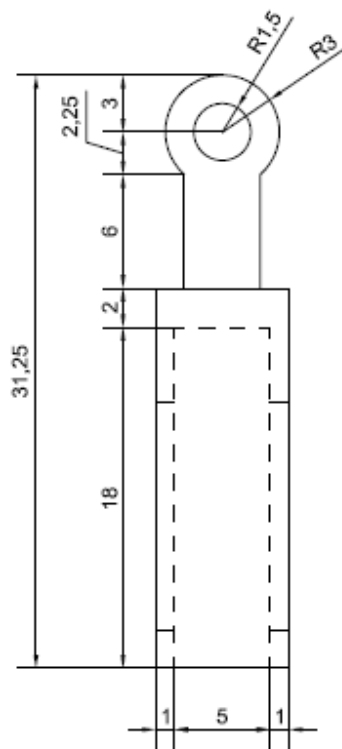
Son un total de 22 planos, divididos en dos: parte delantera y parte trasera, según la zona de la que forman parte, excepto la base, que ha sido metida en la primera sección aunque ella sea la parte central de todo el vehículo.


En caso de alguna duda, se adjuntarán en otra carpeta cada uno de los planos en formato pdf, donde se aprecian mejor los diferentes grosores otorgados a cada una de las partes utilizadas, y los archivos creados de AutoCAD.

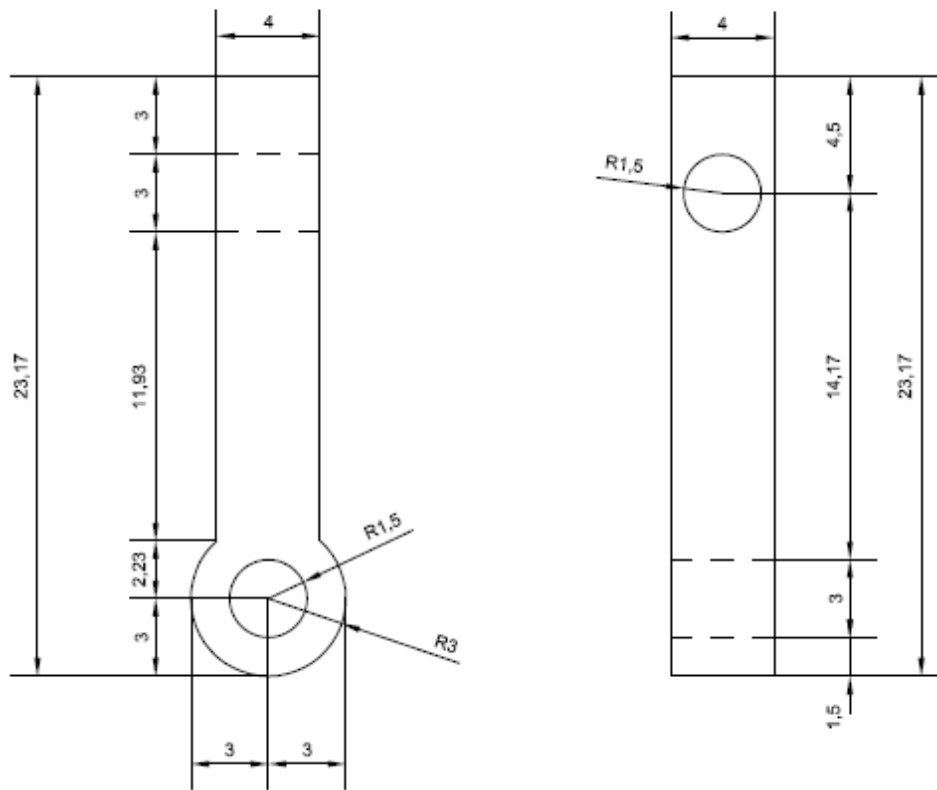
2. PARTE DELANTERA




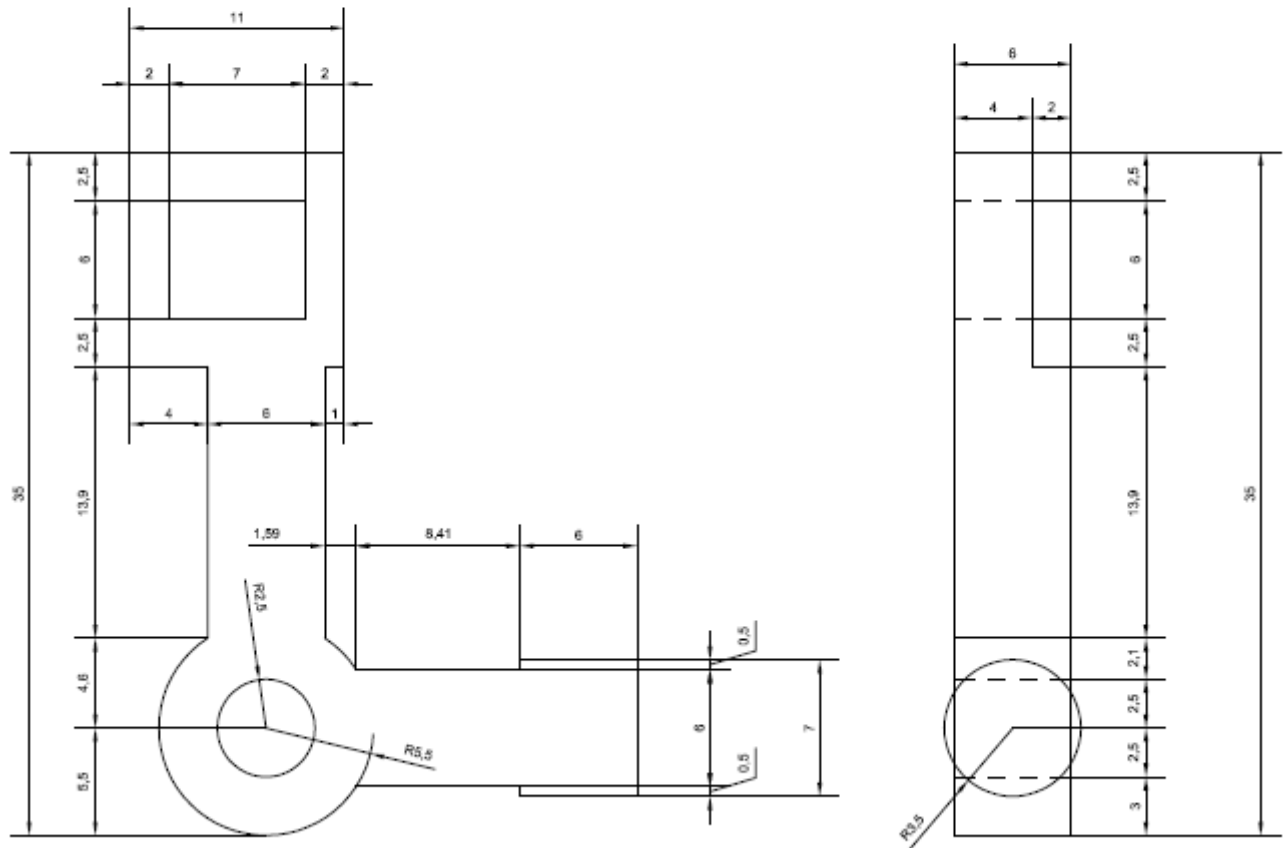
TÍTULO			DISEÑO MECÁNICO, IMPRESIÓN, Y MONTAJE DE UN VEHICULO IMPRIMIBLE		
PLANO			FECHA		
Pieza delantera lateral					
INGENIEROS			PLANO Nº		
Dña. María Ramos Montero			01		
FECHA	PROPIEDAD	ESCALA			
Sept-2012	Universidad Carlos III de Madrid	1:500			



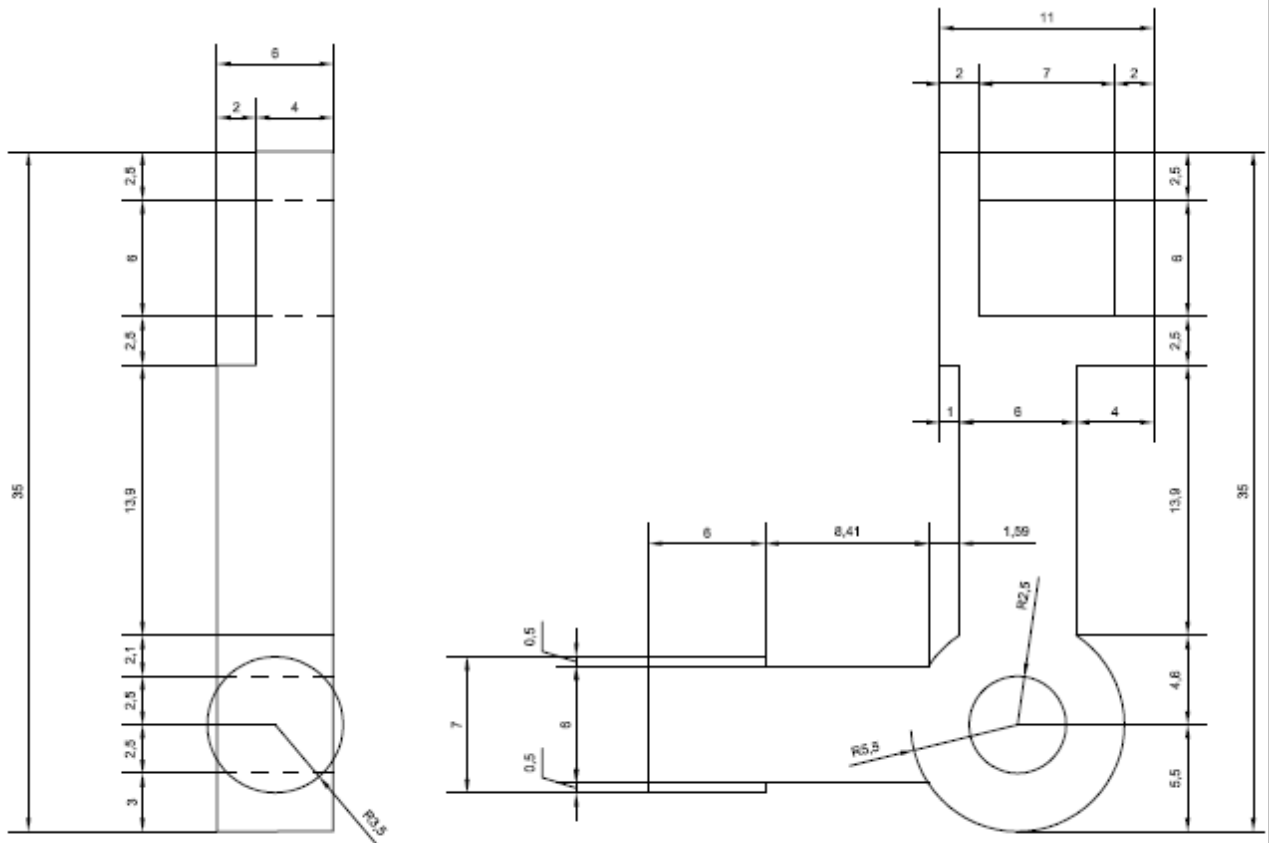
TÍTULO		DISEÑO MECÁNICO. IMPRESIÓN, Y MONTAJE DE UN VEHICULO IMPRIMIBLE	
PLANO		FIRMA	
Suspensión Delantera parte superior			
INGENIEROS		PLANO Nº	
Dña. María Ramos Montero		02	
FECHA	PROPIEDAD	ESCALA	
Sept-2012	Universidad Carlos III de Madrid	1:300	




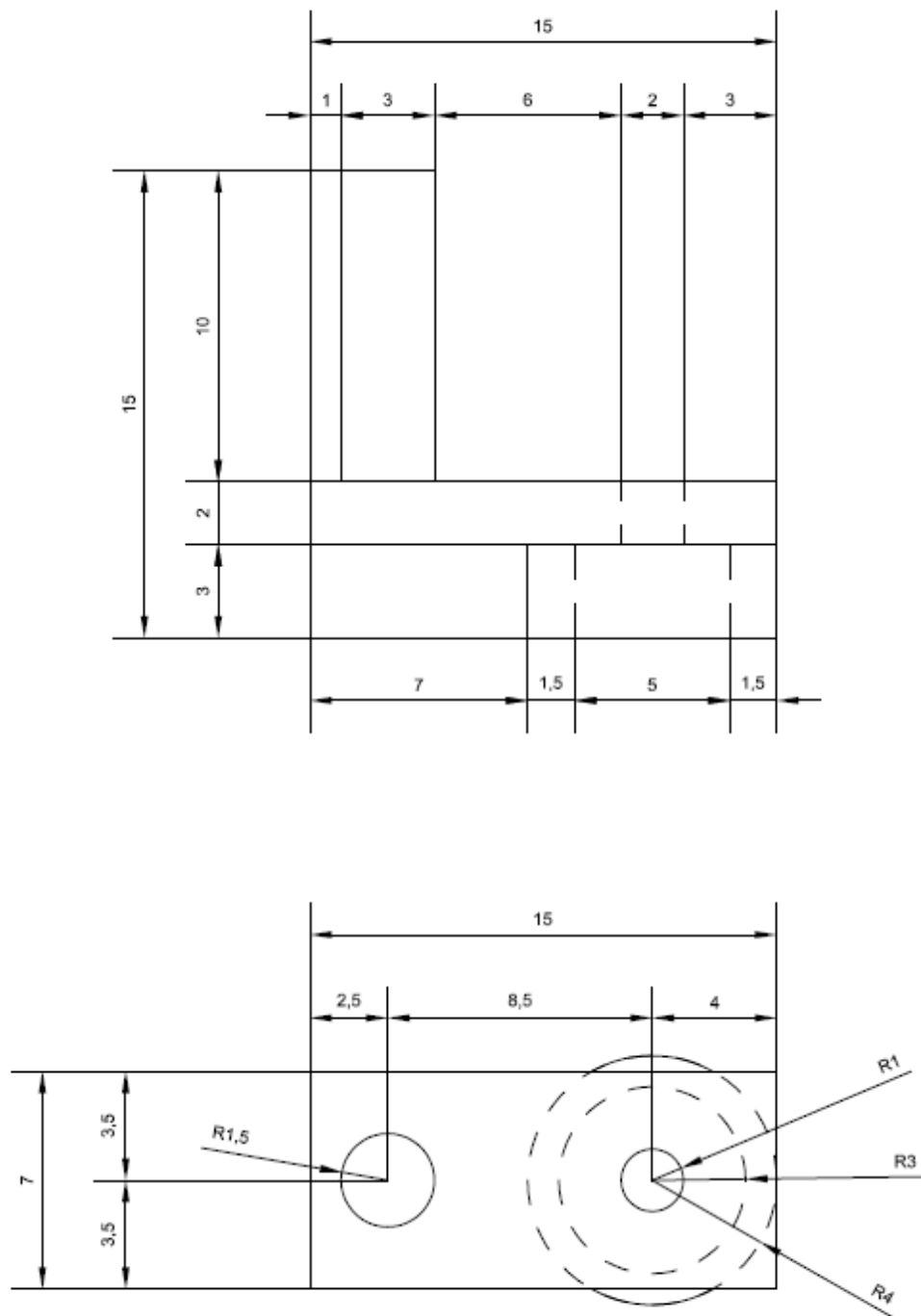
TÍTULO		DISEÑO MECÁNICO, IMPRESIÓN, Y MONTAJE DE UN VEHICULO IMPRIMIBLE	
PLANO		Suspensión Delantera parte Inferior	FECHA 
INGENIEROS		Dña. María Ramos Montero	PLANO Nº 03
FECHA Sept-2012	PROPIEDAD Universidad Carlos III de Madrid	ESCALA 1:250	




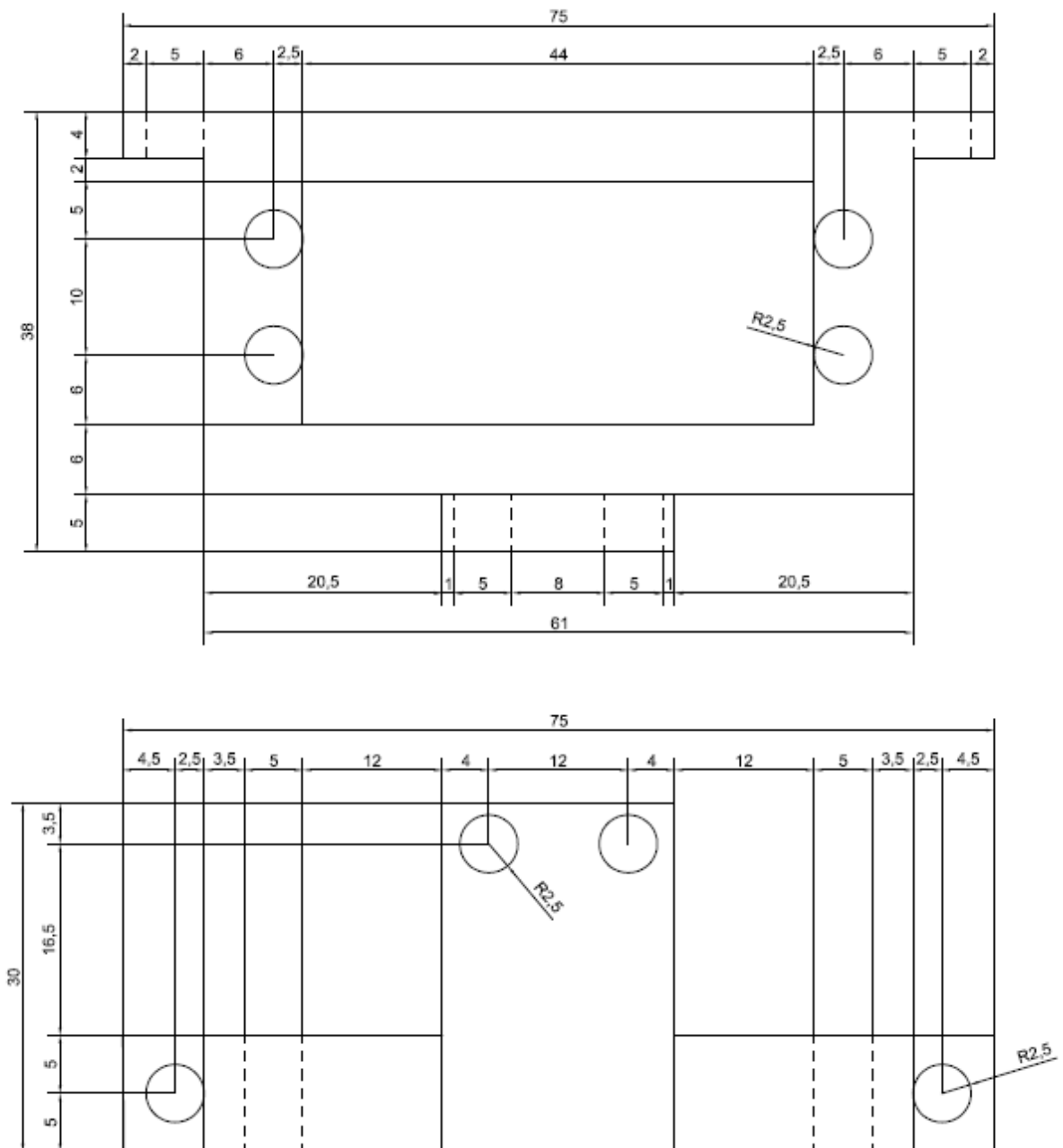
TÍTULO			DISEÑO MECÁNICO, IMPRESIÓN, Y MONTAJE DE UN VEHICULO IMPRIMIBLE
PLANO		Firma	
		Pieza de Dirección Izquierda	
INGENIEROS		PLANO Nº	
		Dña. María Ramos Montero	
		04	
FECHA	PROPIEDAD	ESCALA	
Sept-2012	Universidad Carlos III de Madrid	1:250	



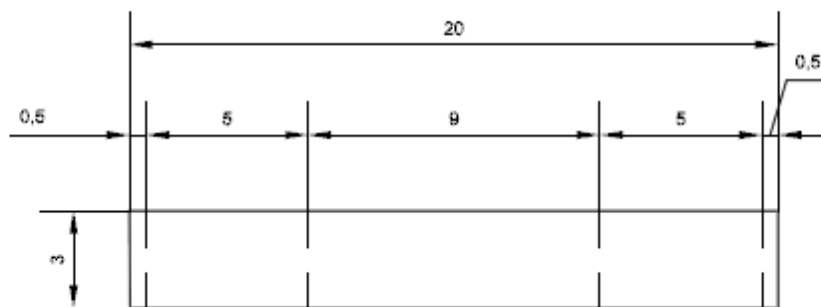
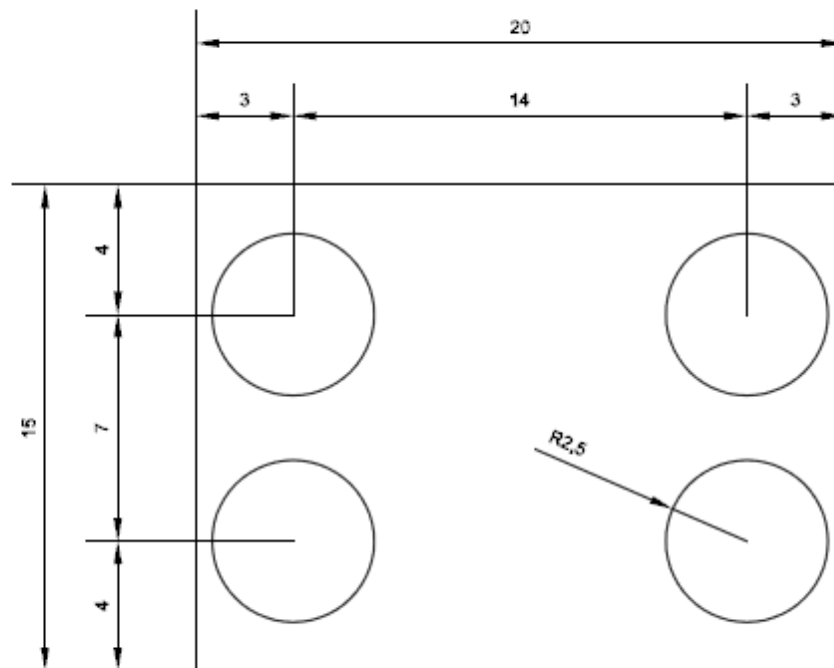
TÍTULO		DISEÑO MECÁNICO, IMPRESIÓN, Y MONTAJE DE UN VEHICULO IMPRIMIBLE	
PLANO		FECHA	
Pieza de Dirección Derecha			
INGENIEROS		PLANO Nº	
Dña. María Ramos Montero		05	
FECHA	PROPIEDAD	ESCALA	
Sept-2012	Universidad Carlos III de Madrid	1:250	



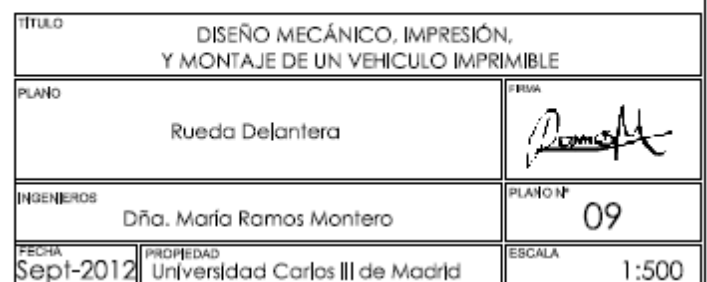
TÍTULO		DISEÑO MECÁNICO, IMPRESIÓN, Y MONTAJE DE UN VEHICULO IMPRIMIBLE	
PLANO		Pieza de Dirección Unida al Servo	FIRMA 
INGENIEROS		Dña. María Ramos Montero	PLANO Nº 06
FECHA	PROPIEDAD	ESCALA	
Sept-2012	Universidad Carlos III de Madrid	1:200	

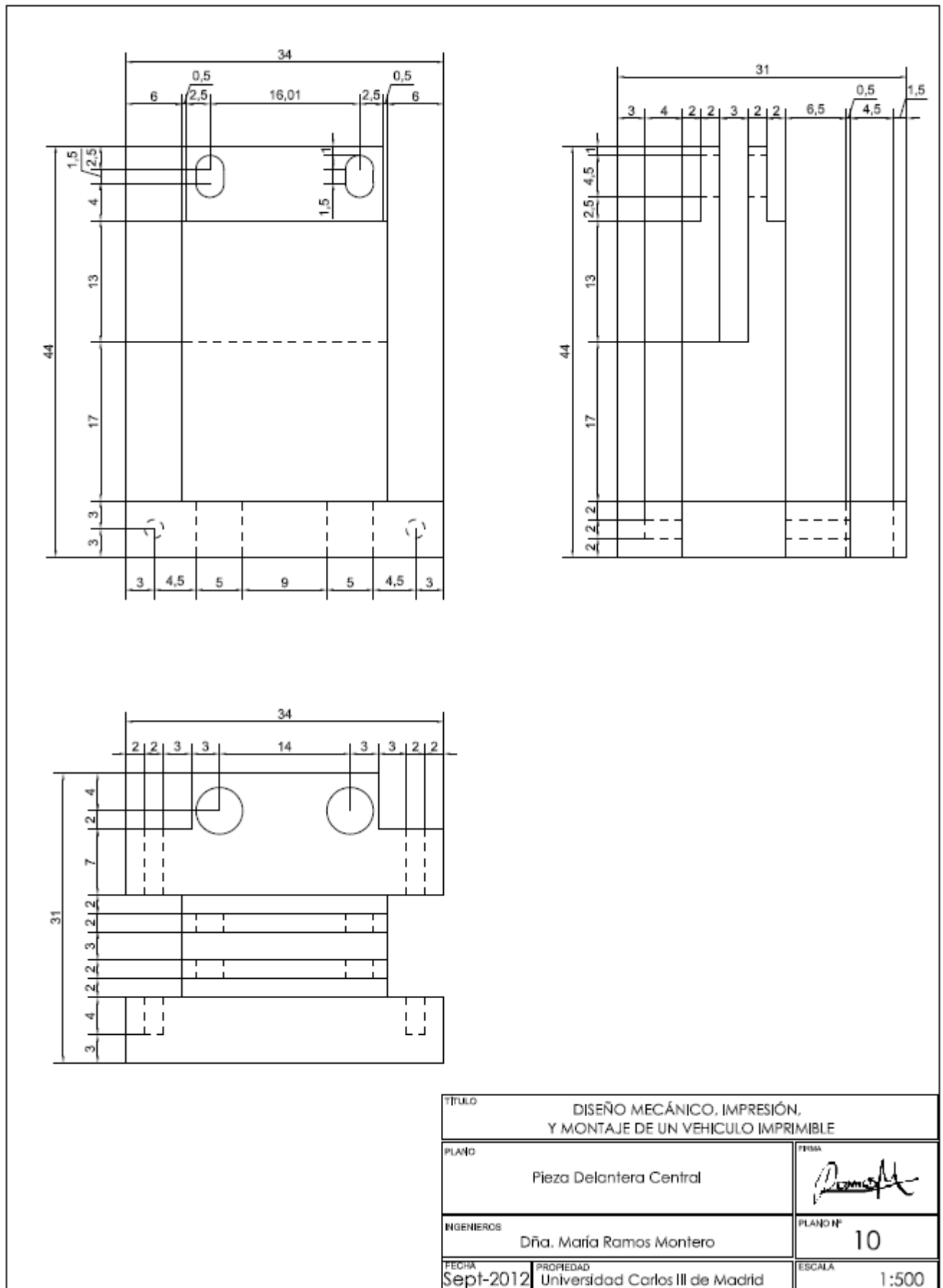


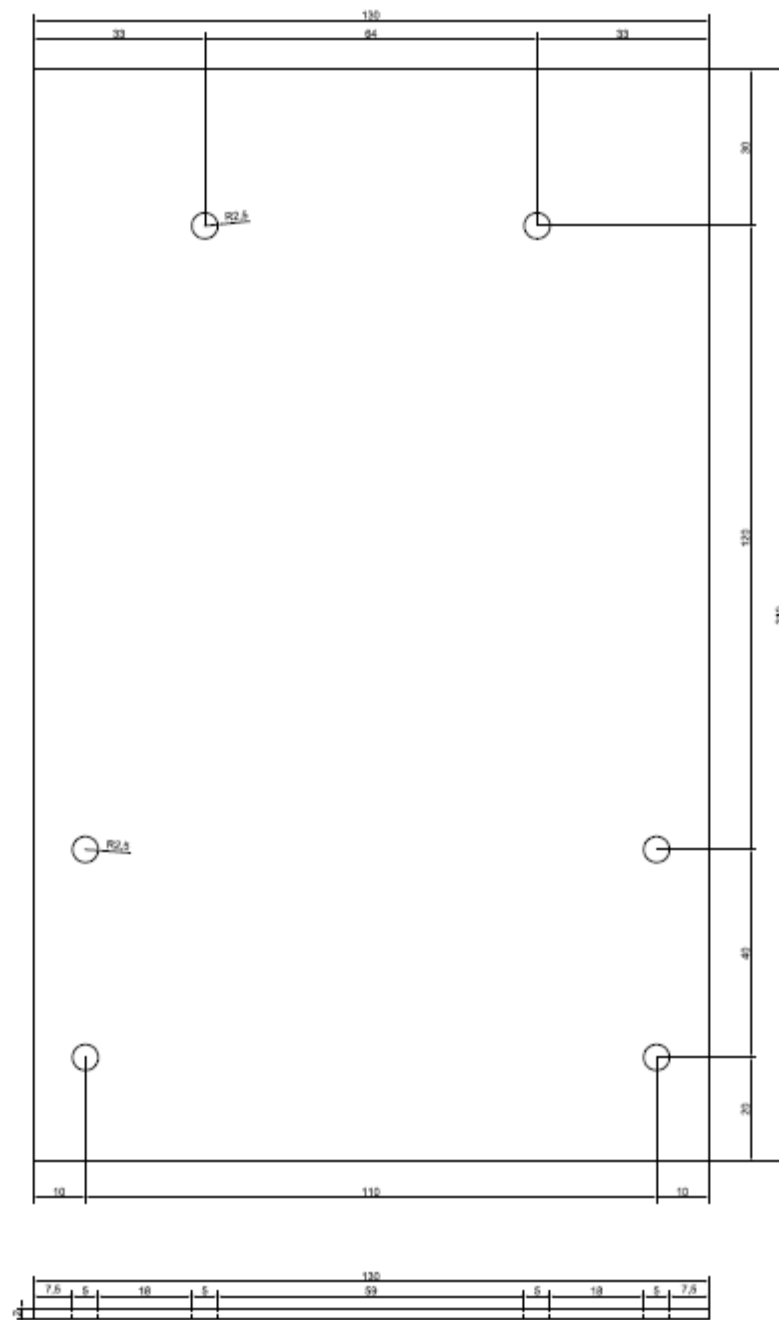
TÍTULO			DISEÑO MECÁNICO, IMPRESIÓN, Y MONTAJE DE UN VEHICULO IMPRIMIBLE
PLANO		Firma	
		Soporte del Servo	
INGENIEROS		PLANO Nº	
Dña. María Ramos Montero		07	
FECHA	PROPIEDAD	ESCALA	
Sept-2012	Universidad Carlos III de Madrid	1:500	



TÍTULO			DISEÑO MECÁNICO, IMPRESIÓN, Y MONTAJE DE UN VEHICULO IMPRIMIBLE
PLANO		FECHA	
Pieza de Unión entre Pieza Delantera Central y Soporte del Servo			
INGENIEROS		PLANO Nº	
Dña. María Ramos Montero		08	
FECHA	PROPIEDAD	ESCALA	
Sept-2012	Universidad Carlos III de Madrid	1:200	

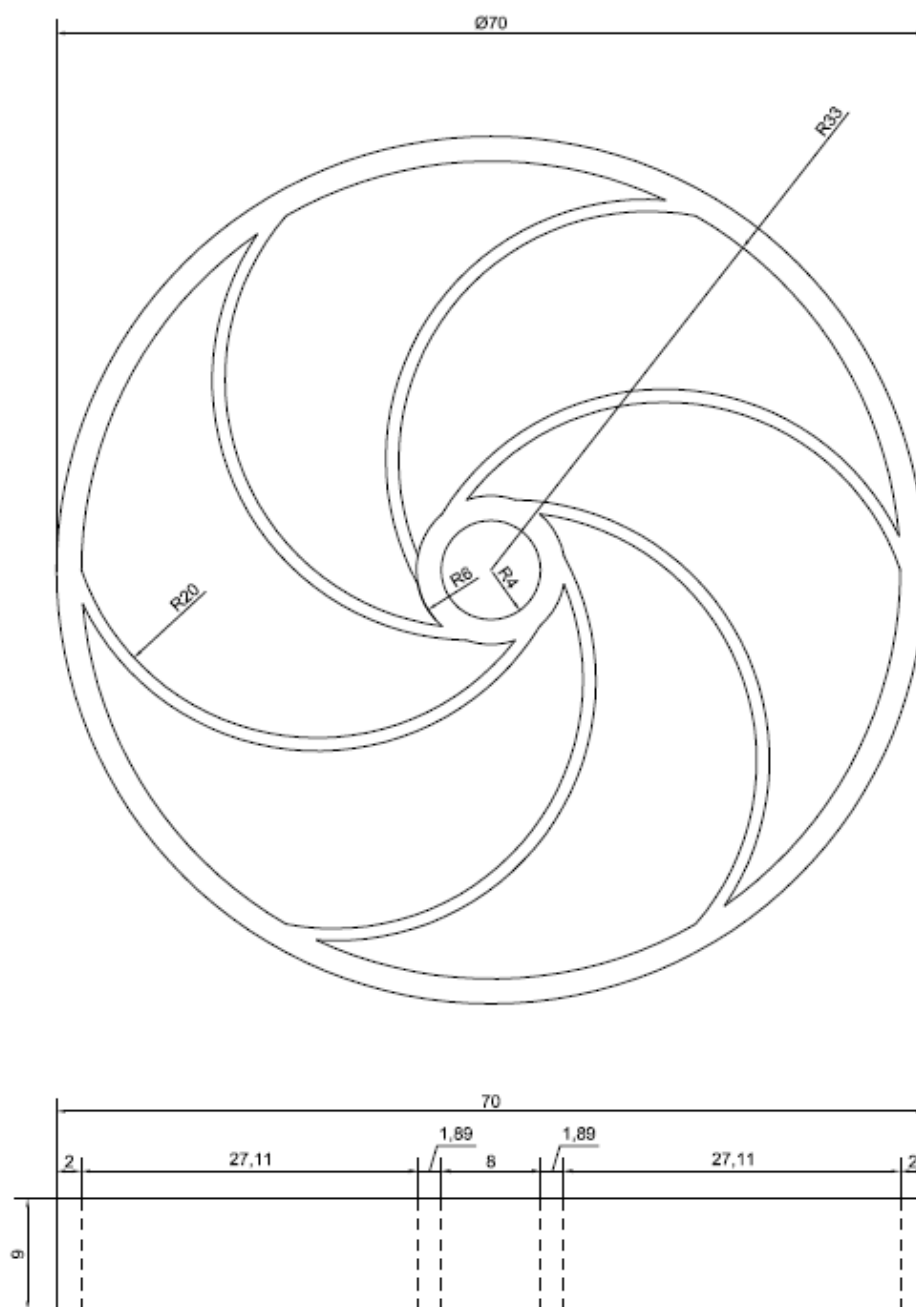





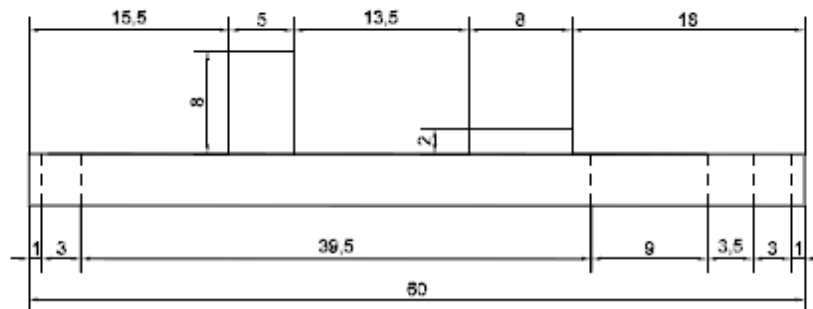
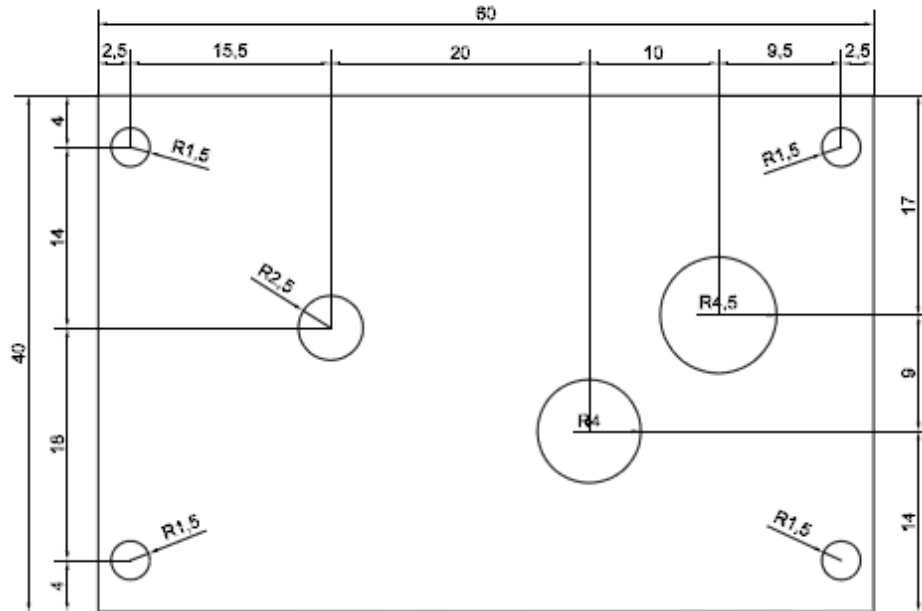



TÍTULO			DISEÑO MECÁNICO, IMPRESIÓN, Y MONTAJE DE UN VEHICULO IMPRIMIBLE
PLANO		FIRMA	
		Placa Base de Metacrilato	
INGENIEROS		PLAN Nº	
		Dña. María Ramos Montero	
FECHA		ESCALA	
Sept-2012		1:750	
PROPIEDAD			
Universidad Carlos III de Madrid			

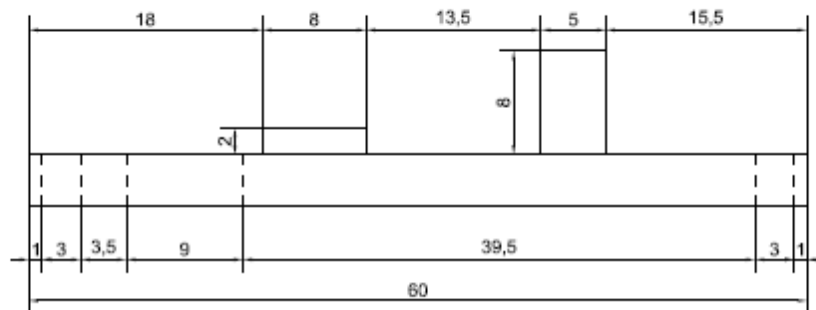
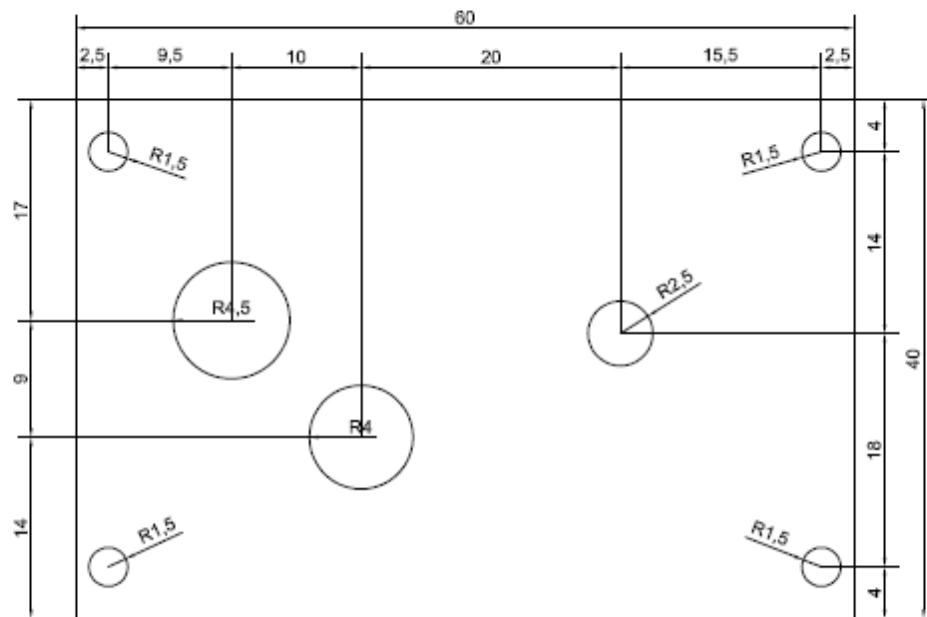
3. PARTE TRASERA




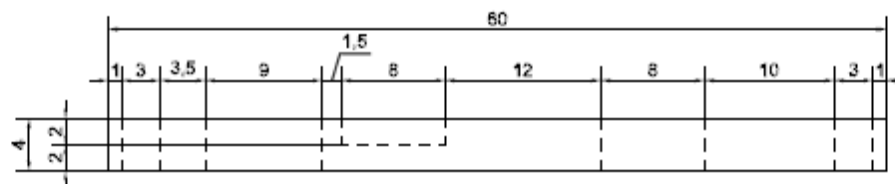
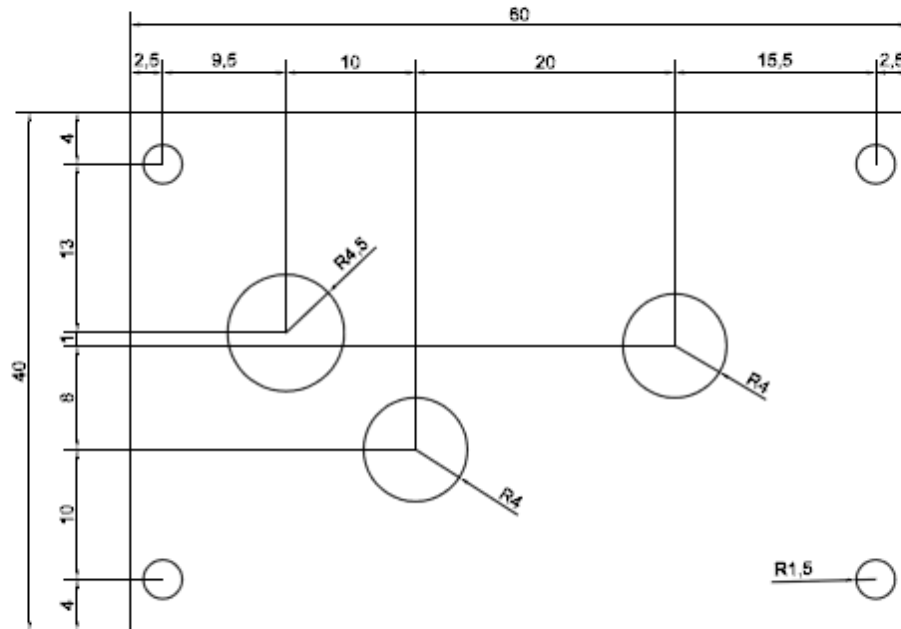
TÍTULO		DISEÑO MECÁNICO, IMPRESIÓN, Y MONTAJE DE UN VEHICULO IMPRIMIBLE	
PLANO		Rueda Trasera con Suspensión	FIRMA 
INGENIEROS		Dña. María Ramos Montero	PLANO Nº 12
FECHA	PROPIEDAD	ESCALA	
Sept-2012	Universidad Carlos III de Madrid	1:500	




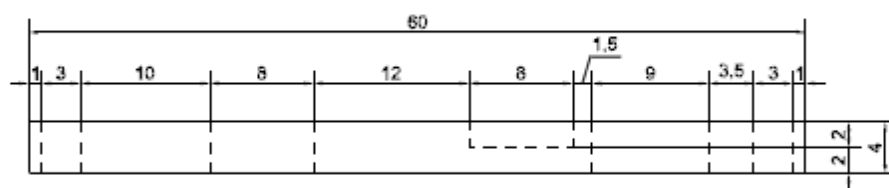
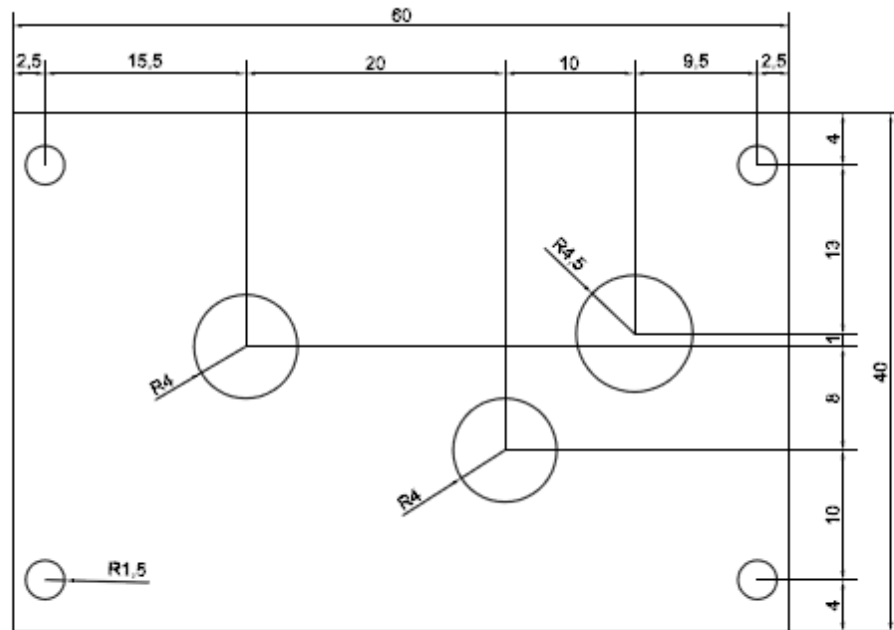
TÍTULO			DISEÑO MECÁNICO, IMPRESIÓN, Y MONTAJE DE UN VEHICULO IMPRIMIBLE
PLANO		Firma	
			
INGENIEROS		PLANO Nº	
Dña. María Ramos Montero		13	
FECHA	PROPIEDAD	ESCALA	
Sept-2012	Universidad Carlos III de Madrid	1:500	




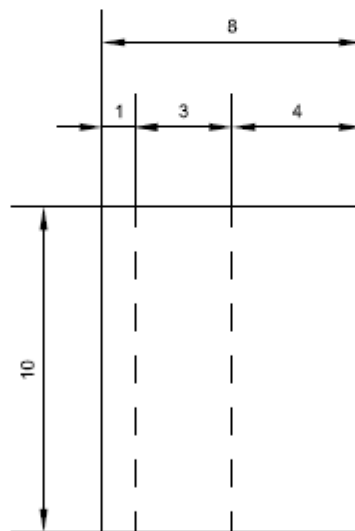
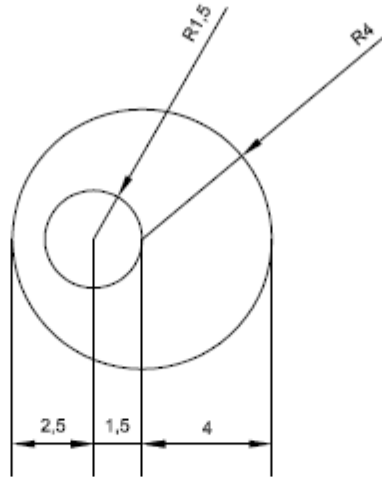
TÍTULO			DISEÑO MECÁNICO, IMPRESIÓN, Y MONTAJE DE UN VEHICULO IMPRIMIBLE
PLANO		firmas	
Soporte Trasero Derecho de los engranajes			
INGENIEROS		PLANO Nº	
Dña. María Ramos Montero		14	
FECHA	PROPIEDAD	ESCALA	
Sept-2012	Universidad Carlos III de Madrid	1:500	




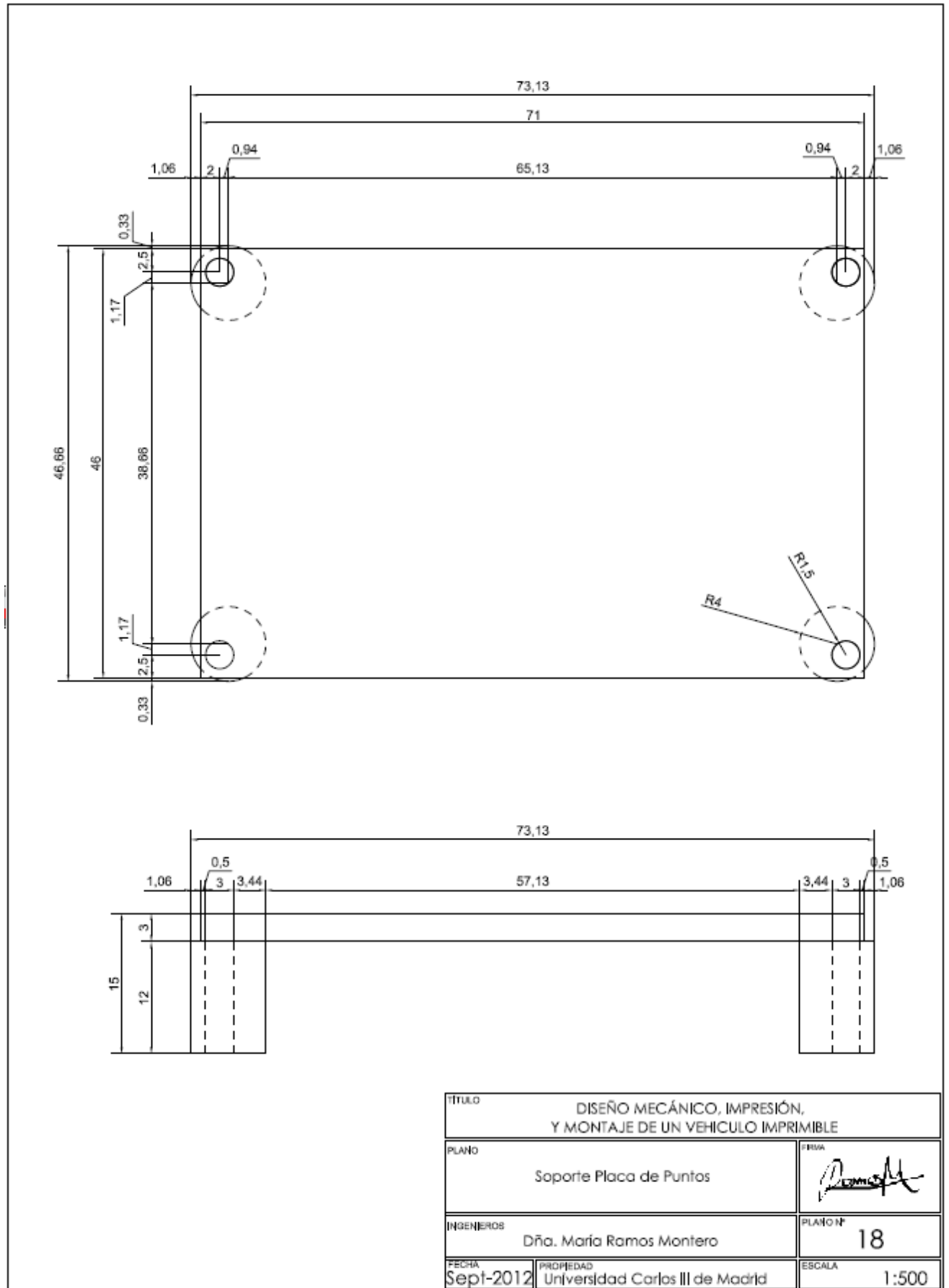
TÍTULO			DISEÑO MECÁNICO, IMPRESIÓN, Y MONTAJE DE UN VEHICULO IMPRIMIBLE
PLANO		FIRMAS	
Soporte Delantero Izquierdo de los engranajes			
INGENIEROS		PLANO Nº	
Dña. María Ramos Montero		15	
FECHA	PROPIEDAD	ESCALA	
Sept-2012	Universidad Carlos III de Madrid	1:500	

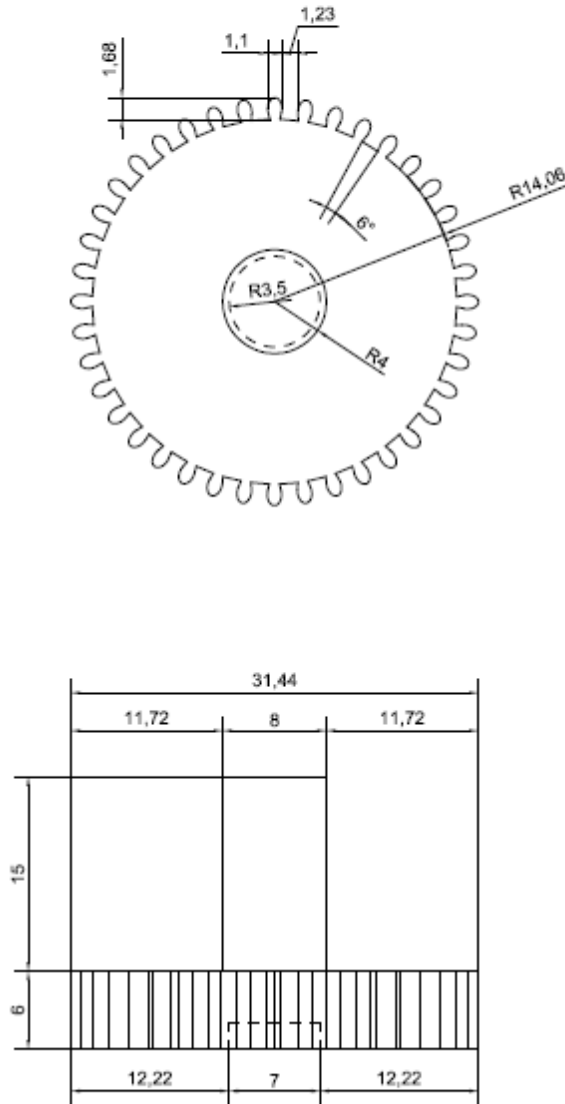



TÍTULO		DISEÑO MECÁNICO, IMPRESIÓN, Y MONTAJE DE UN VEHICULO IMPRIMIBLE	
PLANO		Firma	
Soporte Delantero Derecho de los engranajes			
INGENIEROS		PLANO Nº	
Dña. María Ramos Montero		16	
FECHA	PROPIEDAD	ESCALA	
Sept-2012	Universidad Carlos III de Madrid	1:500	

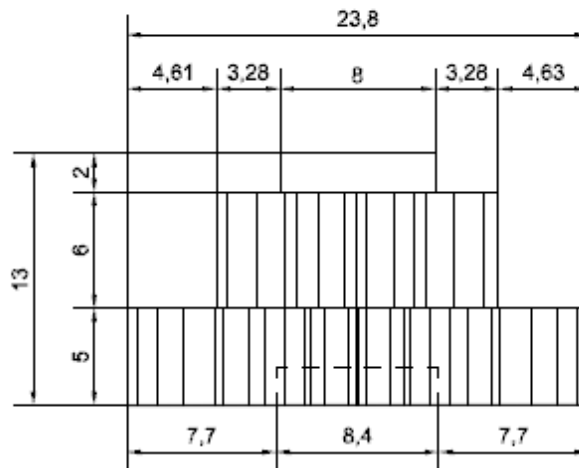
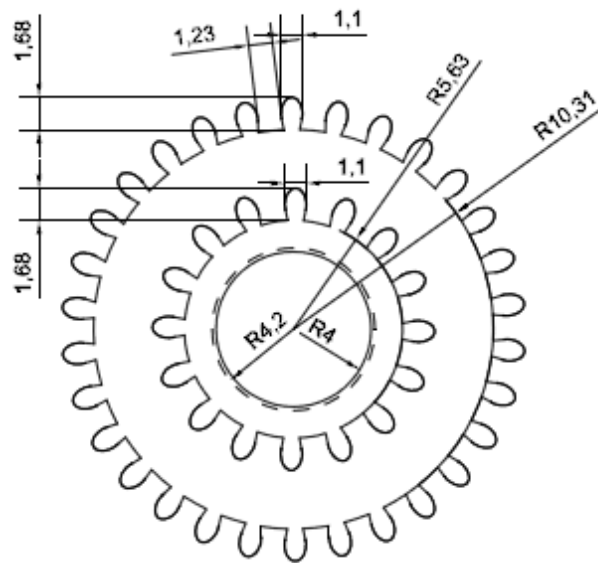


TÍTULO		DISEÑO MECÁNICO, IMPRESIÓN, Y MONTAJE DE UN VEHICULO IMPRIMIBLE	
PLANO		Soporte Placa Arduino	<small>FIRMA</small> 
INGENIEROS		Dña. María Ramos Montero	<small>PLANO Nº</small> 17
<small>FECHA</small>	<small>PROPIEDAD</small>	<small>ESCALA</small>	
Sept-2012	Universidad Carlos III de Madrid	1:200	

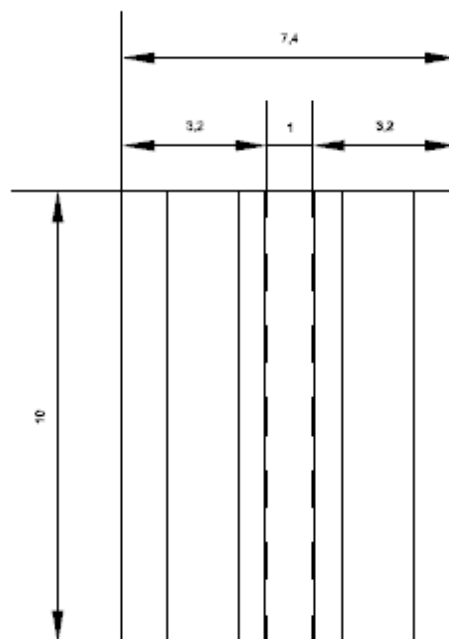
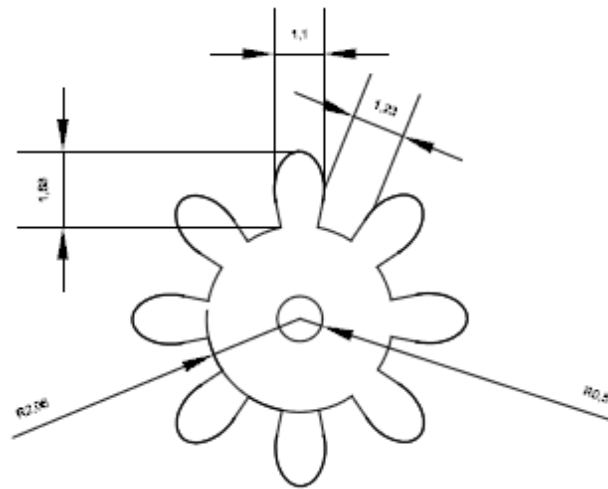




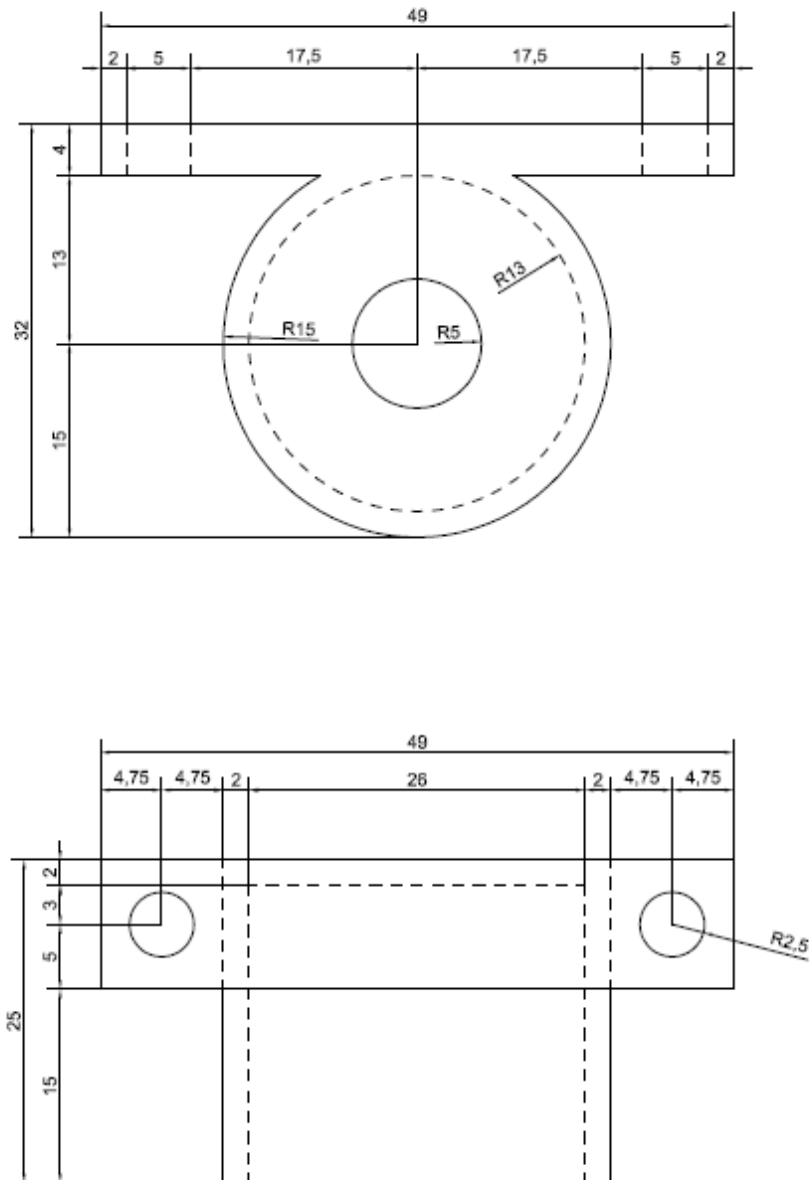
TÍTULO			DISEÑO MECÁNICO, IMPRESIÓN, Y MONTAJE DE UN VEHICULO IMPRIMIBLE
PLANO		PROSA	
INGENIEROS		PLANO Nº	19
FECHA	PROPIEDAD	ESCALA	
Sept-2012	Universidad Carlos III de Madrid	1:500	




TÍTULO		DISEÑO MECÁNICO, IMPRESIÓN, Y MONTAJE DE UN VEHICULO IMPRIMIBLE
PLANO		Engranajes Números 2 y 3
INGENIEROS		Dña. María Ramos Montero
FECHA		Sept-2012
PROPIEDAD		Universidad Carlos III de Madrid
FOLIO		20
ESCALA		1:300



TÍTULO			DISEÑO MECÁNICO, IMPRESIÓN, Y MONTAJE DE UN VEHICULO IMPRIMIBLE
PLANO		Firma	
Engranaje Número 1			
INGENIEROS		PLANO Nº	
Dña. María Ramos Montero		21	
FECHA	PROPIEDAD	ESCALA	
Sept-2012	Universidad Carlos III de Madrid	1:70	



TÍTULO			DISEÑO MECÁNICO, IMPRESIÓN, Y MONTAJE DE UN VEHICULO IMPRIMIBLE		
PLANO			FECHA		
Soporte de los motores					
INGENIEROS			PLANO Nº		
Dña. María Ramos Montero			22		
FECHA		PROPIEDAD		ESCALA	
Sept-2012		Universidad Carlos III de Madrid		1:500	

III. Anexos

A. HOJAS DE CARACTERÍSTICAS

A.1 Motor *rs_540sh*



RS-540SH

Carbon-brush motors

OUTPUT : 5.0W~90W (APPROX)

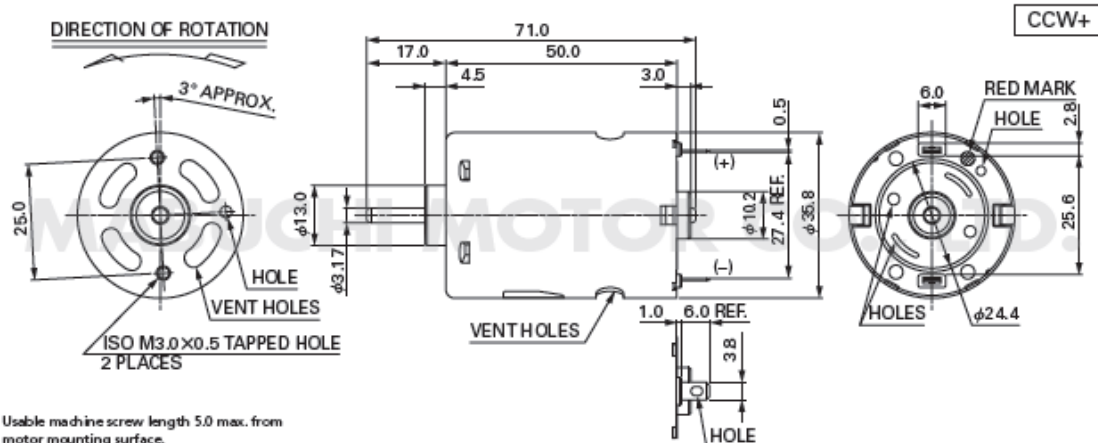
WEIGHT : 160g (APPROX)

Typical Applications Home Appliances : Vacuum Cleaner
Cordless Power Tools : Drill / Screwdriver / Cordless Garden Tool / Air Compressor
Toys and Models : Radio Control Model

MODEL	VOLTAGE		NO LOAD		AT MAXIMUM EFFICIENCY						STALL		
	OPERATING RANGE	NOMINAL	SPEED	CURRENT	SPEED	CURRENT	TORQUE		OUTPUT	TORQUE		CURRENT	
			r/min	A	r/min	A	milin	gcm	W	milin	gcm	A	
RS-540SH-7520 (*1)	4.8~7.2	7.2V CONSTANT	23400	2.40	19740	13.0	30.6	312	63.2	196	1998	70.0	
RS-540SH-6527 (*1)	4.8~9.6	9.6V CONSTANT	23400	1.60	20040	9.55	31.0	316	64.9	216	2202	57.0	

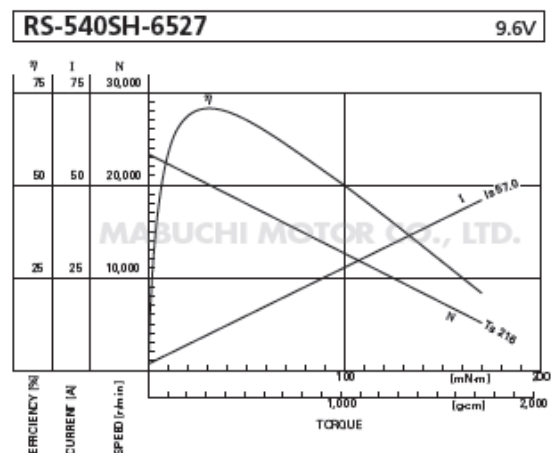
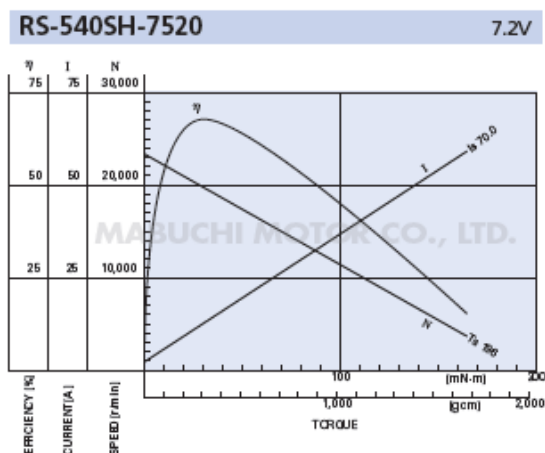
(*1) CCW shifted commutation (CCW+)

The terminal position against the tapped holes varies depending on QW+/NEUTRAL. (Refer to P19)



Usable machine screw length 5.0 max. from motor mounting surface.

UNIT: MILLIMETERS



MABUCHI MOTOR CO., LTD. Headquarters 430 Matsuhidai, Matsudo City, Chiba 270-2280 Japan. Tel:81-47-710-1177 Fax:81-47-710-1132 (Sales Dept.)

A.2 Diodo 1N4001



1N4001 - 1N4007

1.0A RECTIFIER

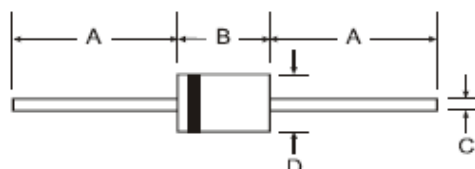
[Please click here to visit our online spice models database.](#)

Features

- Diffused Junction
- High Current Capability and Low Forward Voltage Drop
- Surge Overload Rating to 30A Peak
- Low Reverse Leakage Current
- Lead Free Finish, RoHS Compliant (Note 3)

Mechanical Data

- Case: DO-41
- Case Material: Molded Plastic: UL Flammability Classification Rating 94V-0
- Moisture Sensitivity: Level 1 per J-STD-020D
- Terminals: Finish - Bright Tin. Plated Leads Solderable per MIL-STD-202, Method 208
- Polarity: Cathode Band
- Mounting Position: Any
- Ordering Information: See Page 2
- Marking: Type Number
- Weight: 0.30 grams (approximate)



Dim	DO-41 Plastic	
	Min	Max
A	25.40	—
B	4.06	5.21
C	0.71	0.864
D	2.00	2.72
All Dimensions in mm		

Maximum Ratings and Electrical Characteristics @T_A = 25°C unless otherwise specified

Single phase, half wave, 60Hz, resistive or inductive load.
For capacitive load, derate current by 20%.

Characteristic	Symbol	1N4001	1N4002	1N4003	1N4004	1N4005	1N4006	1N4007	Unit	
Peak Repetitive Reverse Voltage	V _{RRM}									
Working Peak Reverse Voltage	V _{RWM}	50	100	200	400	600	800	1000	V	
DC Blocking Voltage	V _R									
RMS Reverse Voltage	V _{R(RMS)}	35	70	140	280	420	560	700	V	
Average Rectified Output Current (Note 1) @ T _A = 75°C	I _O	1.0							A	
Non-Repetitive Peak Forward Surge Current 8.3ms single half sine-wave superimposed on rated load	I _{FSM}	30							A	
Forward Voltage @ I _F = 1.0A	V _{FM}	1.0							V	
Peak Reverse Current @T _A = 25°C	I _{RM}	5.0							μA	
at Rated DC Blocking Voltage @ T _A = 100°C		50								
Typical Junction Capacitance (Note 2)	C _j	15				8				pF
Typical Thermal Resistance Junction to Ambient	R _{θJA}	100							K/W	
Maximum DC Blocking Voltage Temperature	T _A	+150							°C	
Operating and Storage Temperature Range	T _J , T _{STG}	-65 to +150							°C	

- Notes:
1. Leads maintained at ambient temperature at a distance of 9.5mm from the case.
 2. Measured at 1.0 MHz and applied reverse voltage of 4.0V DC.
 3. EU Directive 2002/95/EC (RoHS). All applicable RoHS exemptions applied, see EU Directive 2002/95/EC Annex Notes.

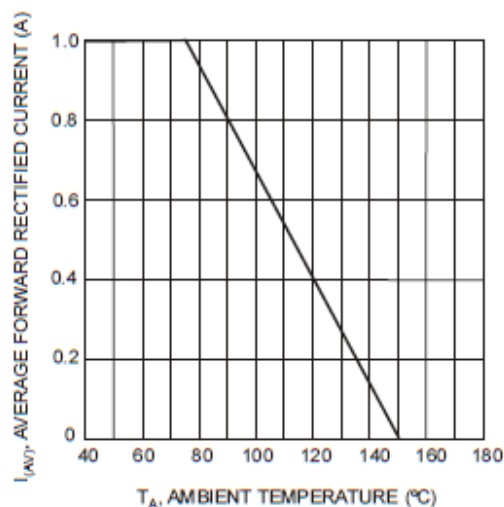


Fig. 1 Forward Current Derating Curve

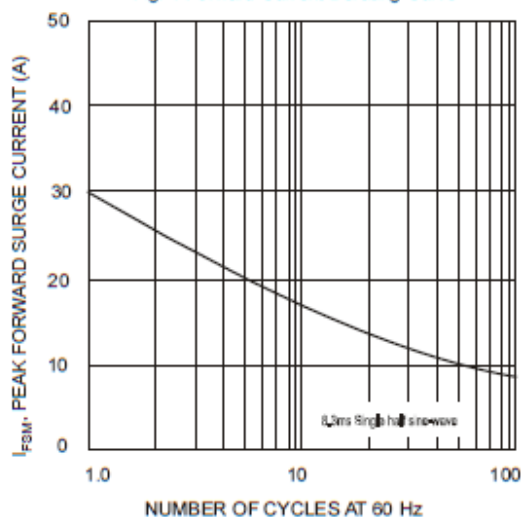


Fig. 3 Max Non-Repetitive Peak Fwd Surge Current

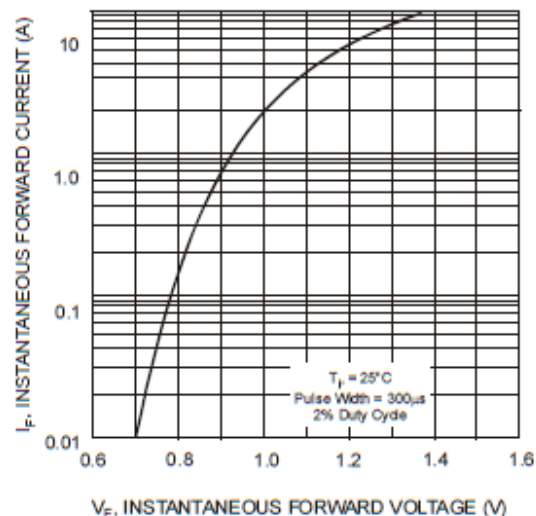


Fig. 2 Typical Forward Characteristics

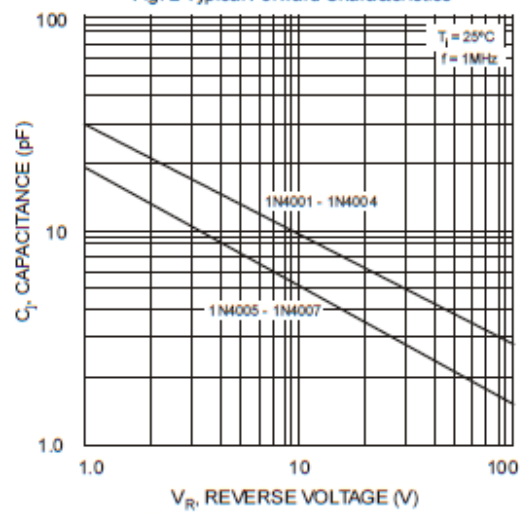


Fig. 4 Typical Junction Capacitance

Ordering Information (Note 4)

Device	Packaging	Shipping
1N4001-B	DO-41 Plastic	1K/Bulk
1N4001-T	DO-41 Plastic	5K/Tape & Reel, 13-inch
1N4002-B	DO-41 Plastic	1K/Bulk
1N4002-T	DO-41 Plastic	5K/Tape & Reel, 13-inch
1N4003-B	DO-41 Plastic	1K/Bulk
1N4003-T	DO-41 Plastic	5K/Tape & Reel, 13-inch
1N4004-B	DO-41 Plastic	1K/Bulk
1N4004-T	DO-41 Plastic	5K/Tape & Reel, 13-inch
1N4005-B	DO-41 Plastic	1K/Bulk
1N4005-T	DO-41 Plastic	5K/Tape & Reel, 13-inch
1N4006-B	DO-41 Plastic	1K/Bulk
1N4006-T	DO-41 Plastic	5K/Tape & Reel, 13-inch
1N4007-B	DO-41 Plastic	1K/Bulk
1N4007-T	DO-41 Plastic	5K/Tape & Reel, 13-inch

Notes: 4. For packaging details, visit our website at <http://www.diodes.com/datasheets/ap02008.pdf>.

A.3 Circuito Integrado L293B



SGS-THOMSON
MICROELECTRONICS

L293B
L293E

PUSH-PULL FOUR CHANNEL DRIVERS

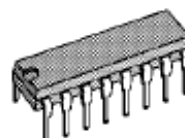
- OUTPUT CURRENT 1A PER CHANNEL
- PEAK OUTPUT CURRENT 2A PER CHANNEL (non repetitive)
- INHIBIT FACILITY
- HIGH NOISE IMMUNITY
- SEPARATE LOGIC SUPPLY
- OVERTEMPERATURE PROTECTION

DESCRIPTION

The L293B and L293E are quad push-pull drivers capable of delivering output currents to 1A per channel. Each channel is controlled by a TTL-compatible logic input and each pair of drivers (a full bridge) is equipped with an inhibit input which turns off all four transistors. A separate supply input is provided for the logic so that it may be run off a lower voltage to reduce dissipation.

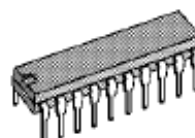
Additionally, the L293E has external connection of sensing resistors, for switchmode control.

The L293B and L293E are package in 16 and 20-pin plastic DIPs respectively ; both use the four center pins to conduct heat to the printed circuit board.



DIP16

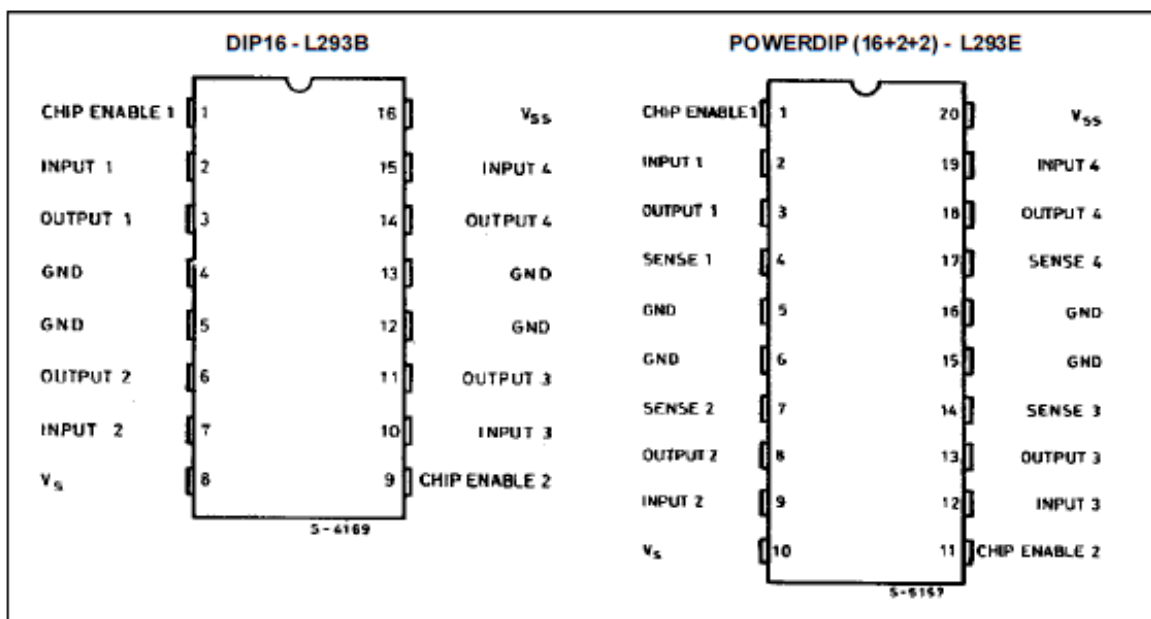
ORDERING NUMBER : L293B



POWERDIP (16 + 2 + 2)

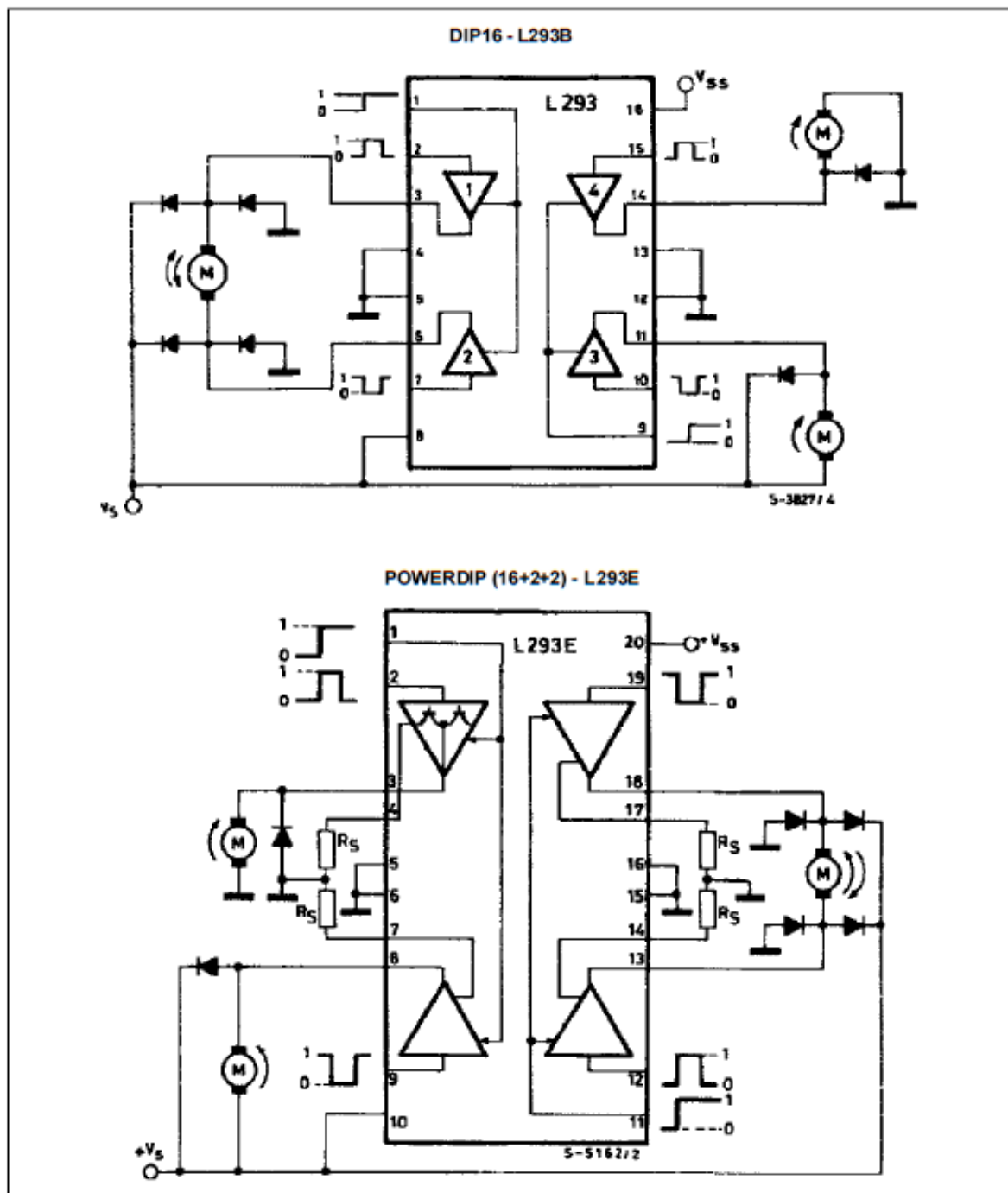
ORDERING NUMBER : L293E

PIN CONNECTIONS



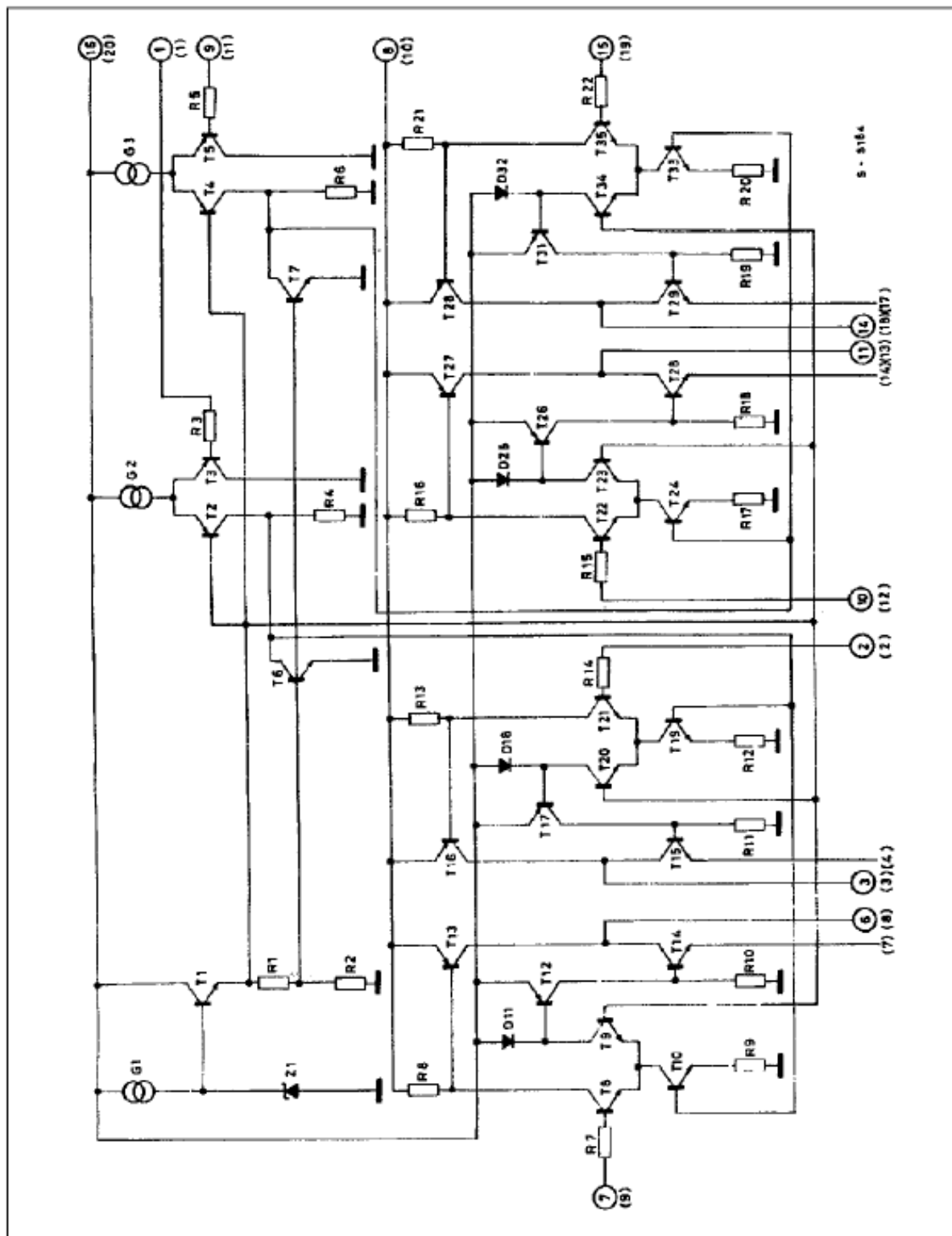
L293B - L293E

BLOCK DIAGRAMS



L293B - L293E

SCHEMATIC DIAGRAM



(*) In the L293 these points are not externally available. They are internally connected to the ground (substrate).

O Pins of L293

() Pins of L293E.

L293B - L293E**ABSOLUTE MAXIMUM RATINGS**

Symbol	Parameter	Value	Unit
V_S	Supply Voltage	36	V
V_{SS}	Logic Supply Voltage	36	V
V_i	Input Voltage	7	V
V_{inh}	Inhibit Voltage	7	V
I_{out}	Peak Output Current (non repetitive $t = 5ms$)	2	A
P_{tot}	Total Power Dissipation at $T_{ground-pins} = 80^\circ C$	5	W
T_{stg}, T_j	Storage and Junction Temperature	-40 to +150	$^\circ C$

THERMAL DATA

Symbol	Parameter	Value	Unit
$R_{thj-case}$	Thermal Resistance Junction-case	Max. 14	$^\circ C/W$
$R_{thj-amb}$	Thermal Resistance Junction-ambient	Max. 80	$^\circ C/W$

ELECTRICAL CHARACTERISTICS

For each channel, $V_S = 24V$, $V_{SS} = 5V$, $T_{amb} = 25^\circ C$, unless otherwise specified

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
V_S	Supply Voltage		V_{SS}		36	V
V_{SS}	Logic Supply Voltage		4.5		36	V
I_S	Total Quiescent Supply Current	$V_i = L \quad I_o = 0 \quad V_{inh} = H$ $V_i = H \quad I_o = 0 \quad V_{inh} = H$ $V_{inh} = L$		2 16	6 24 4	mA
I_{SS}	Total Quiescent Logic Supply Current	$V_i = L \quad I_o = 0 \quad V_{inh} = H$ $V_i = H \quad I_o = 0 \quad V_{inh} = H$ $V_{inh} = L$		44 16 16	60 22 24	mA
V_{iL}	Input Low Voltage		-0.3		1.5	V
V_{iH}	Input High Voltage	$V_{SS} \leq 7V$ $V_{SS} > 7V$	2.3 2.3		V_{SS} 7	V
I_{iL}	Low Voltage Input Current	$V_{iL} = 1.5V$			-10	μA
I_{iH}	High Voltage Input Current	$2.3V \leq V_{iH} \leq V_{SS} - 0.6V$		30	100	μA
V_{inhL}	Inhibit Low Voltage		-0.3		1.5	V
V_{inhH}	Inhibit High Voltage	$V_{SS} \leq 7V$ $V_{SS} > 7V$	2.3 2.3		V_{SS} 7	V
I_{inhL}	Low Voltage Inhibit Current	$V_{inhL} = 1.5V$		-30	-100	μA
I_{inhH}	High Voltage Inhibit Current	$2.3V \leq V_{inhH} \leq V_{SS} - 0.6V$			± 10	μA
V_{CEsatH}	Source Output Saturation Voltage	$I_o = -1A$		1.4	1.8	V
V_{CEsatL}	Sink Output Saturation Voltage	$I_o = 1A$		1.2	1.8	V
V_{SENS}	Sensing Voltage (pins 4, 7, 14, 17) (**)				2	V
t_r	Rise Time	0.1 to $0.9 V_o$ (*)		250		ns
t_f	Fall Time	0.9 to $0.1 V_o$ (*)		250		ns
t_{on}	Turn-on Delay	$0.5 V_i$ to $0.5 V_o$ (*)		750		ns
t_{off}	Turn-off Delay	$0.5 V_i$ to $0.5 V_o$ (*)		200		ns

* See figure 1

** Referred to L293E

TRUTH TABLE

V_i (each channel)	V_o	$V_{inh}^{(**)}$
H	H	H
L	L	H
H	X ^(*)	L
L	X ^(*)	L

(*) High output impedance

(**) Relative to the considerate channel

Figure 1 : Switching Timers

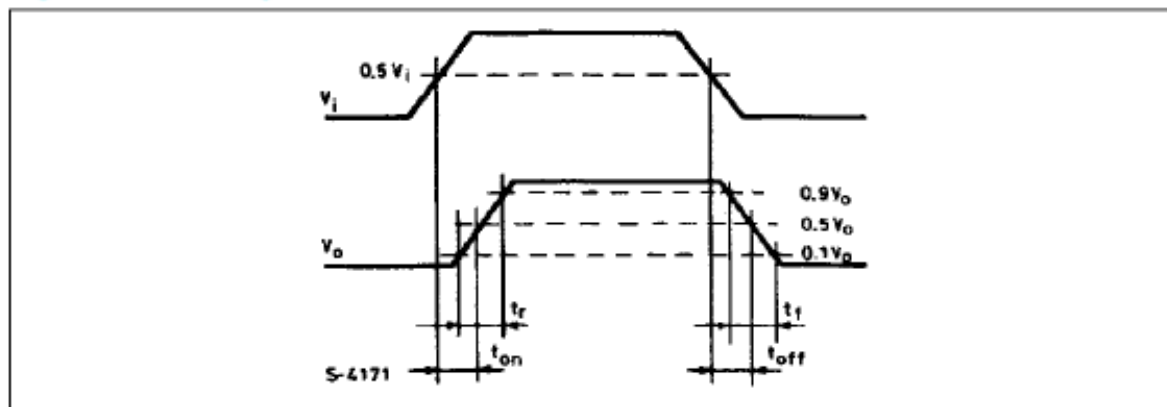


Figure 2 : Saturation voltage versus Output Current

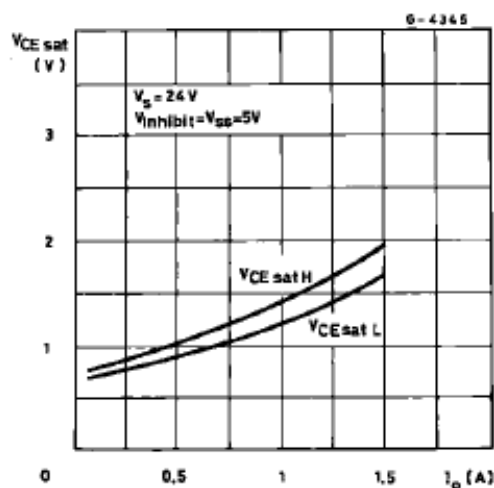


Figure 4 : Sink Saturation Voltage versus Ambient Temperature

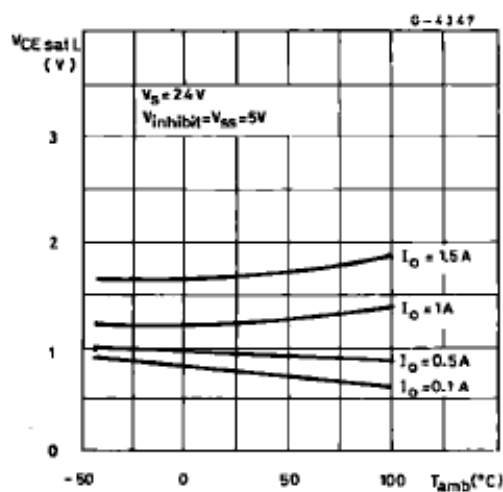


Figure 3 : Source Saturation Voltage versus Ambient Temperature

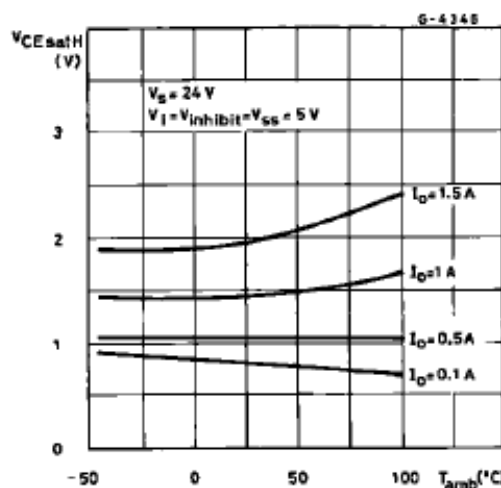
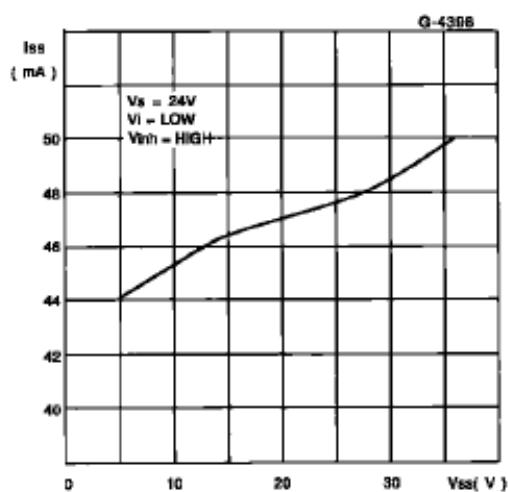


Figure 5 : Quiescent Logic Supply Current versus Logic Supply Voltage



L293B - L293E

Figure 6 : Output Voltage versus Input Voltage

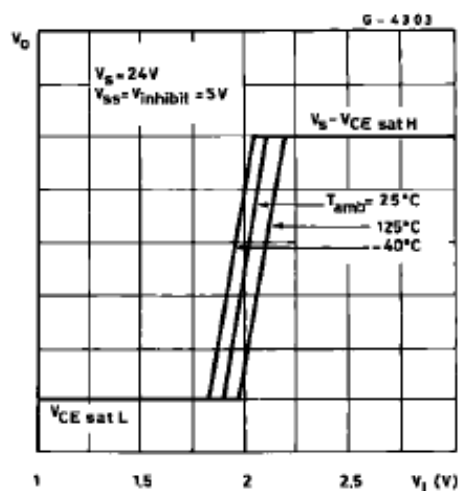
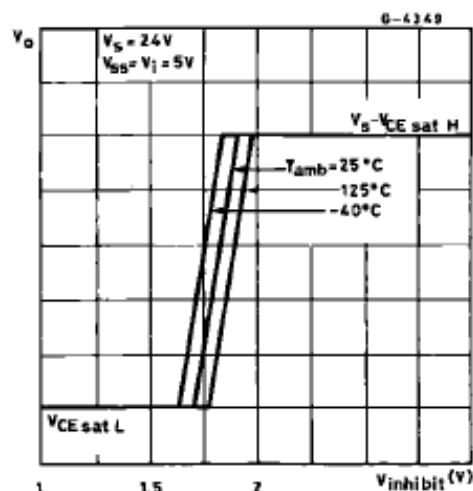
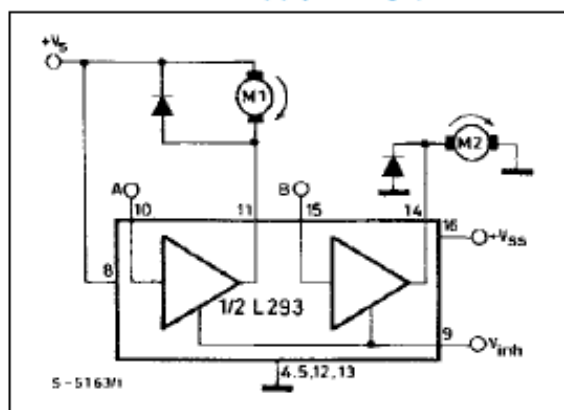


Figure 7 : Output Voltage versus Inhibit Voltage



APPLICATION INFORMATION

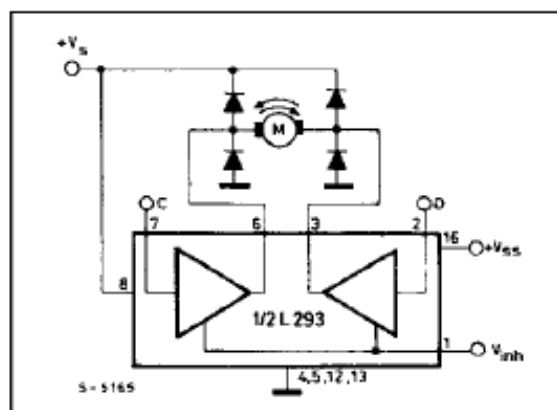
Figure 8 : DC Motor Controls
(with connection to ground and to the supply voltage)



V_{inh}	A	M1	B	M2
H	H	Fast Motor Stop	H	Run
H	L	Run	L	Fast Motor Stop
L	X	Free Running Motor Stop	X	Free Running Motor Stop

L = Low H = High X = Don't Care

Figure 9 : Bidirectional DC Motor Control

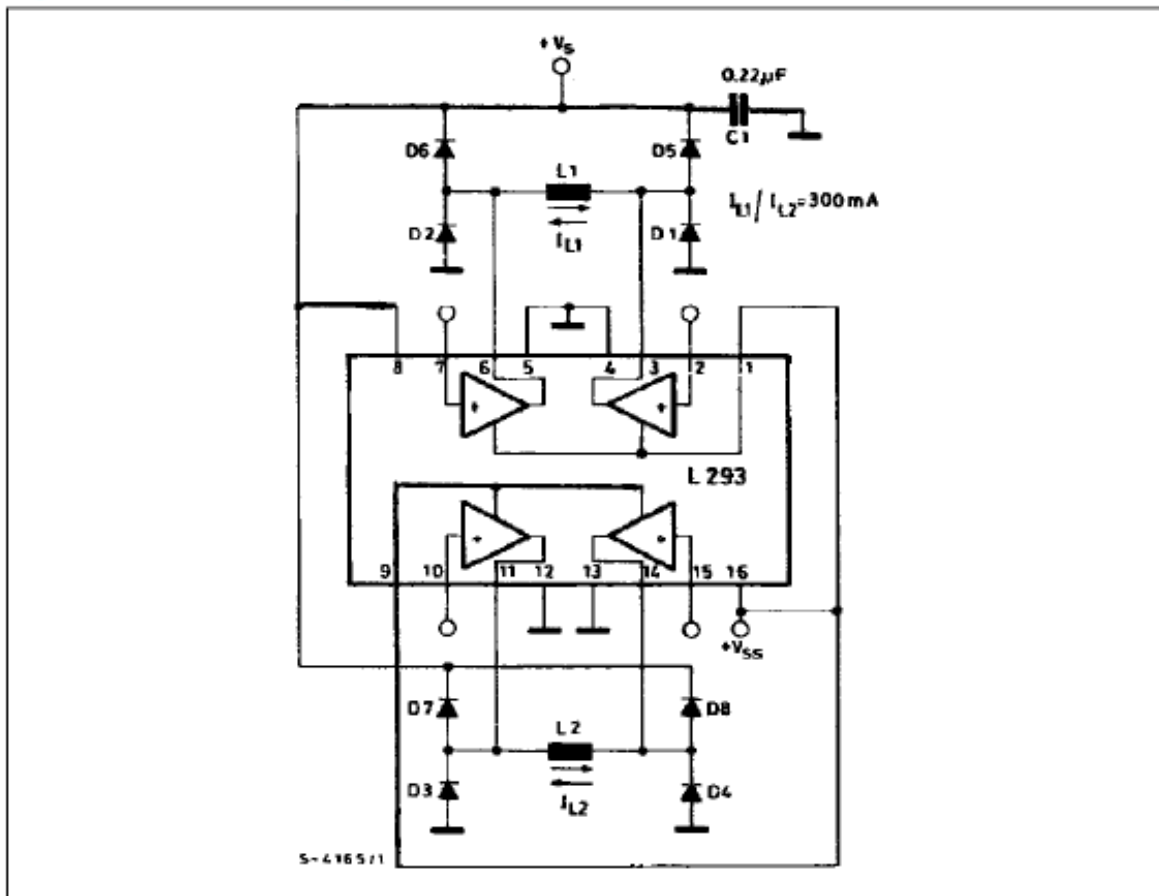


Inputs	Function	
$V_{inh} = H$	C = H ; D = L	Turn Right
	C = L ; D = H	Turn Left
	C = D	Fast Motor Stop
$V_{inh} = L$	C = X ; D = X	Free Running Motor Stop

L = Low H = High X = Don't Care

L293B - L293E

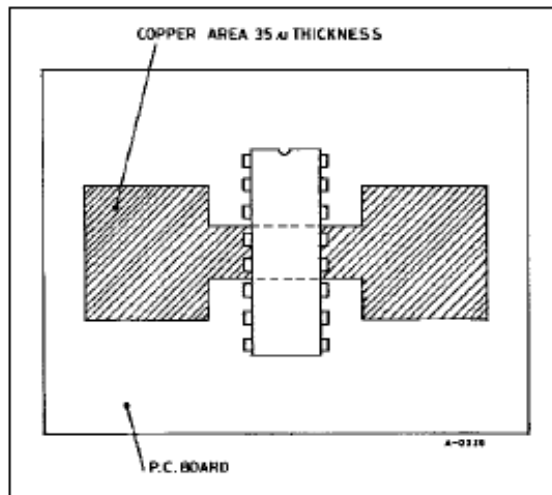
Figure 10 : Bipolar Stepping Motor Control



L293B - L293E**MOUNTING INSTRUCTIONS**

The $R_{th j-amb}$ of the L293B and the L293E can be reduced by soldering the GND pins to a suitable copper area of the printed circuit board as shown in figure 12 or to an external heatsink (figure 13).

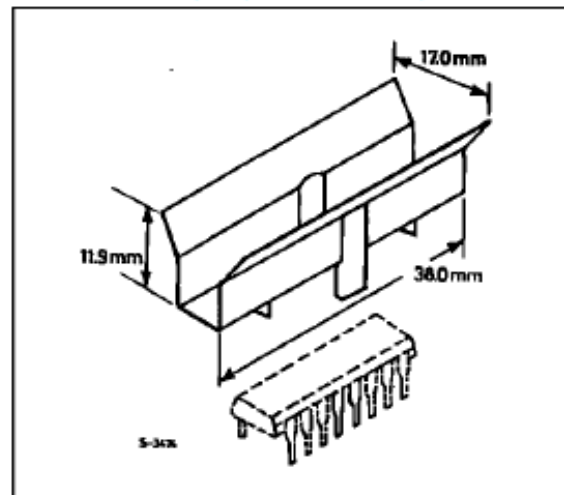
Figure 12 :Example of P.C. Board Copper Area which is Used as Heatsink



During soldering the pins temperature must not exceed 260°C and the soldering time must not be longer than 12 seconds.

The external heatsink or printed circuit copper area must be connected to electrical ground.

Figure 13 :External Heatsink Mounting Example ($R_{th} = 30^{\circ}\text{C/W}$)

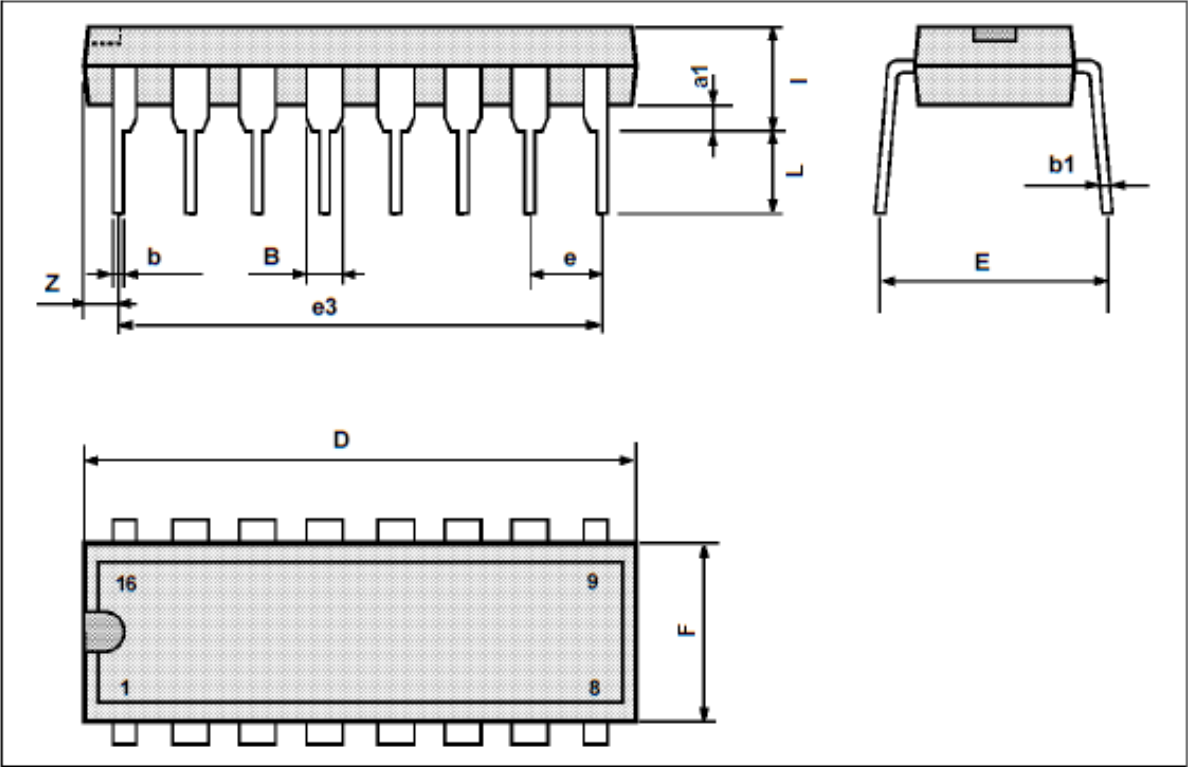


L293B - L293E

DIP16 PACKAGE MECHANICAL DATA

Dimensions	Millimeters			Inches		
	Min.	Typ.	Max.	Min.	Typ.	Max.
a1	0.51			0.020		
B	0.77		1.65	0.030		0.065
b		0.5			0.020	
b1		0.25			0.010	
D			20			0.787
E		8.5			0.335	
e		2.54			0.100	
e3		17.78			0.700	
F			7.1			0.280
i			5.1			0.201
L		3.3			0.130	
Z			1.27			0.050

DIP16PW/TBL



PMDP16N/EP5

A.4 Atmega32u4

Features

- High Performance, Low Power AVR[®] 8-Bit Microcontroller
- Advanced RISC Architecture
 - 135 Powerful Instructions – Most Single Clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 16 MIPS Throughput at 16 MHz
 - On-Chip 2-cycle Multiplier
- Non-volatile Program and Data Memories
 - 16/32K Bytes of In-System Self-Programmable Flash (ATmega16U4/ATmega32U4)
 - 1.25/2.5K Bytes Internal SRAM (ATmega16U4/ATmega32U4)
 - 512Bytes/1K Bytes Internal EEPROM (ATmega16U4/ATmega32U4)
 - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
 - Data retention: 20 years at 85°C/ 100 years at 25°C⁽¹⁾
 - Optional Boot Code Section with Independent Lock Bits
 - In-System Programming by On-chip Boot Program
 - True Read-While-Write Operation
 - All supplied parts are preprogrammed with a default USB bootloader
 - Programming Lock for Software Security
- JTAG (IEEE std. 1149.1 compliant) Interface
 - Boundary-scan Capabilities According to the JTAG Standard
 - Extensive On-chip Debug Support
 - Programming of Flash, EEPROM, Fuses, and Lock Bits through the JTAG Interface
- USB 2.0 Full-speed/Low Speed Device Module with Interrupt on Transfer Completion
 - Complies fully with Universal Serial Bus Specification Rev 2.0
 - Supports data transfer rates up to 12 Mbit/s and 1.5 Mbit/s
 - Endpoint 0 for Control Transfers: up to 64-bytes
 - 6 Programmable Endpoints with IN or Out Directions and with Bulk, Interrupt or Isochronous Transfers
 - Configurable Endpoints size up to 256 bytes in double bank mode
 - Fully independent 832 bytes USB DPRAM for endpoint memory allocation
 - Suspend/Resume Interrupts
 - CPU Reset possible on USB Bus Reset detection
 - 48 MHz from PLL for Full-speed Bus Operation
 - USB Bus Connection/Disconnection on Microcontroller Request
 - Crystal-less operation for Low Speed mode
- Peripheral Features
 - On-chip PLL for USB and High Speed Timer: 32 up to 96 MHz operation
 - One 8-bit Timer/Counter with Separate Prescaler and Compare Mode
 - Two 16-bit Timer/Counter with Separate Prescaler, Compare- and Capture Mode
 - One 10-bit High-Speed Timer/Counter with PLL (64 MHz) and Compare Mode
 - Four 8-bit PWM Channels
 - Four PWM Channels with Programmable Resolution from 2 to 16 Bits
 - Six PWM Channels for High Speed Operation, with Programmable Resolution from 2 to 11 Bits
 - Output Compare Modulator
 - 12-channels, 10-bit ADC (features Differential Channels with Programmable Gain)
 - Programmable Serial USART with Hardware Flow Control
 - Master/Slave SPI Serial Interface



8-bit AVR[®]
Microcontroller
 with
16/32K Bytes of
ISP Flash
and USB
Controller

ATmega16U4
ATmega32U4

Preliminary
Summary

7766FS-AVR-11/10



ATmega16/32U4

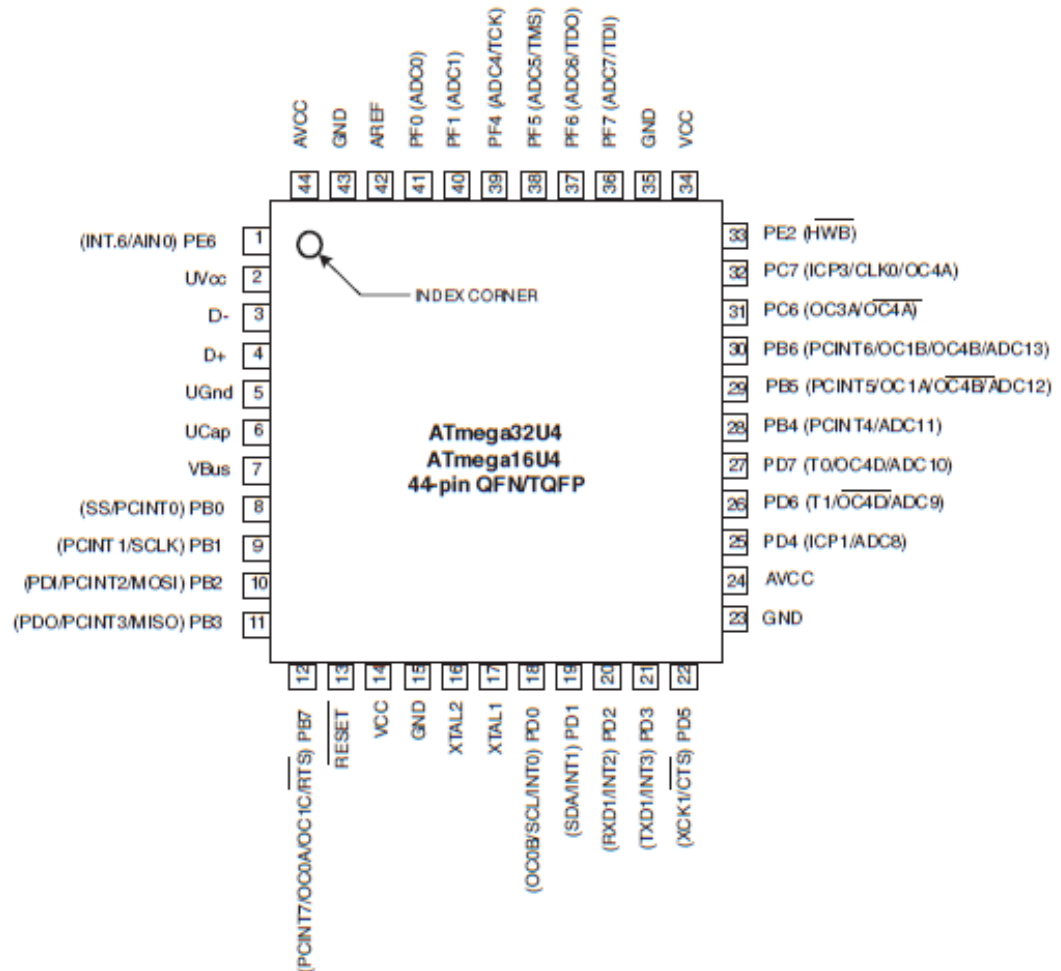
- Byte Oriented 2-wire Serial Interface
- Programmable Watchdog Timer with Separate On-chip Oscillator
- On-chip Analog Comparator
- Interrupt and Wake-up on Pin Change
- On-chip Temperature Sensor
- Special Microcontroller Features
 - Power-on Reset and Programmable Brown-out Detection
 - Internal 8 MHz Calibrated Oscillator
 - Internal clock prescaler & On-the-fly Clock Switching (Int RC / Ext Osc)
 - External and Internal Interrupt Sources
 - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby, and Extended Standby
- I/O and Packages
 - All I/O combine CMOS outputs and LVTTL inputs
 - 26 Programmable I/O Lines
 - 44-lead TQFP Package, 10x10mm
 - 44-lead QFN Package, 7x7mm
- Operating Voltages
 - 2.7 - 5.5V
- Operating temperature
 - Industrial (-40° C to +85° C)
- Maximum Frequency
 - 8 MHz at 2.7V - Industrial range
 - 16 MHz at 4.5V - Industrial range

Note: 1. See "Data Retention" on page 8 for details.

ATmega16/32U4

1. Pin Configurations

Figure 1-1. Pinout ATmega16U4/ATmega32U4



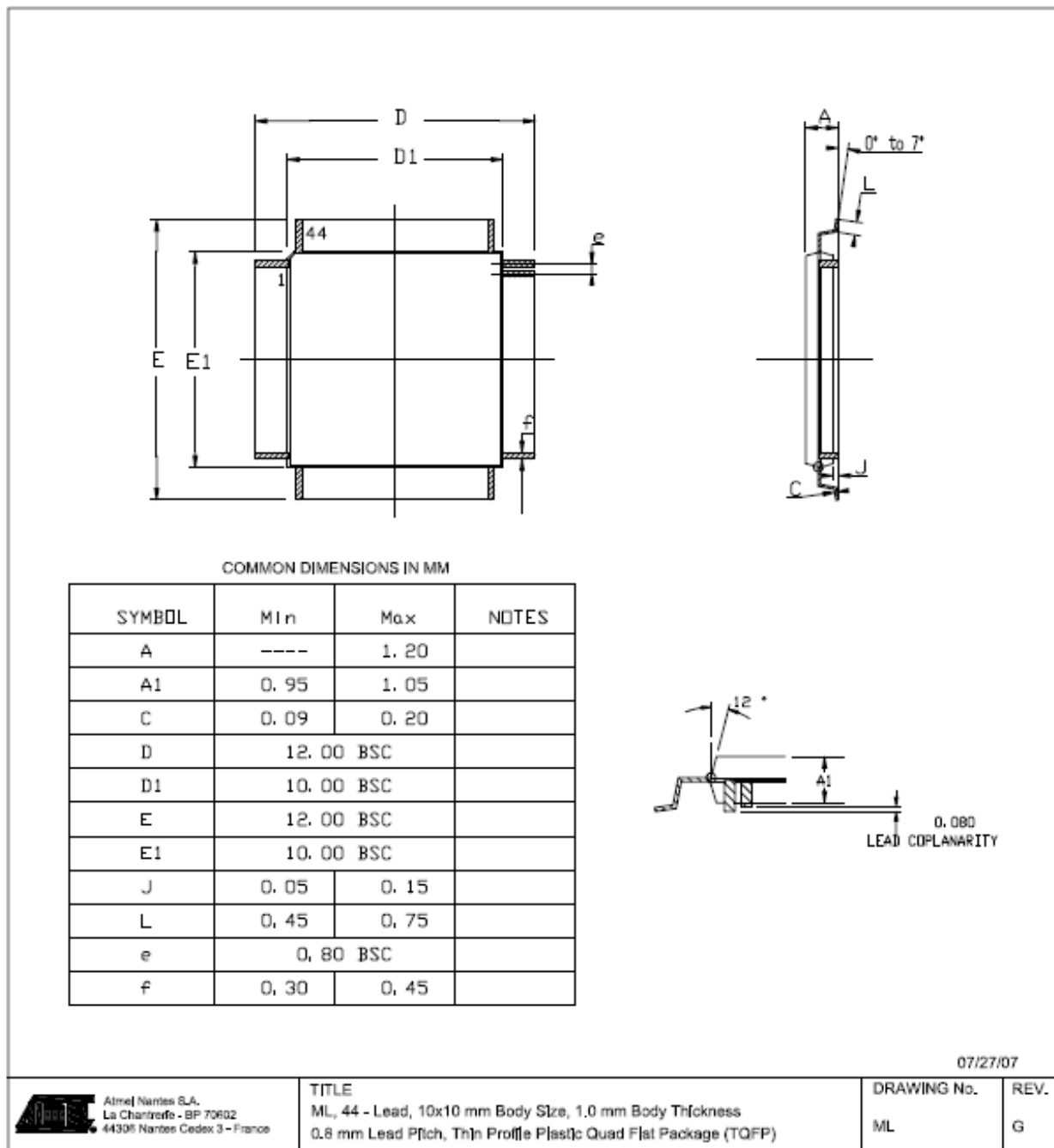
2. Overview

The ATmega16U4/ATmega32U4 is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega16U4/ATmega32U4 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

ATmega16/32U4

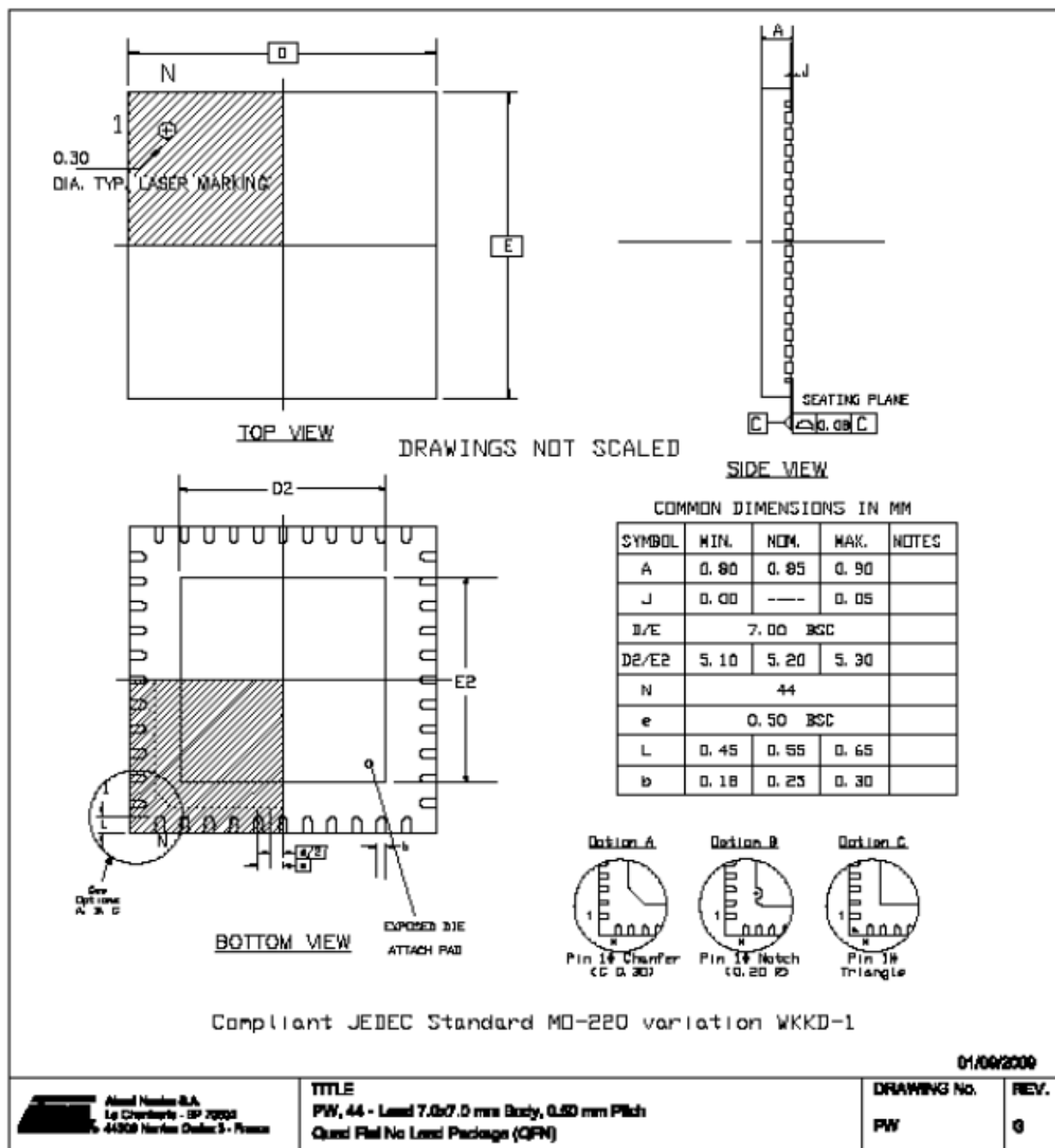
7. Packaging Information

7.1 TQFP44



ATmega16/32U4

7.2 QFN44



B. PLANIFICACIÓN DE TAREAS Y PRESUPUESTO

B.1 Organización de Tareas

En este apartado vamos a hacer un esquema de las tareas llevadas a cabo para la confección del proyecto, con la duración requerida en cada una.

La ejecución de este proyecto ha sido llevada a cabo durante 10 meses en la Universidad Carlos III de Madrid. La realización del mismo se divide en las siguientes tareas:

1. Formación

Este punto se realizó en el curso extraescolar, en el cual se estudia el software y hardware que se requieren para el uso de las impresoras, que permite a su vez el acceso a ellas, además de otros recursos necesarios para crear robots imprimibles.

- **Duración:** 5 meses
- **Tiempo empleado:** 2h/semana

Dentro de este punto, lo podemos dividir en las diferentes enseñanzas aprendidas y usadas para poder llevar a cabo el proyecto:

a. Aprendizaje OpenScad

- **Objetivo:** Instruirse en el uso de los comandos del programa para conseguir diseñar piezas 3D con él.

b. Aprendizaje OOML2

- **Objetivo:** Aprender el manejo de los comandos de C++ para conseguir diseñar piezas 3D con él y obtenerlas en OpenScad.

c. Aprendizaje Replicatorg

- **Objetivo:** Instruirse en el empleo del programa para poder realizar la impresión de los diseños realizados con los recursos anteriores.

d. Aprendizaje de lenguaje Arduino

- **Objetivo:** Aprender el uso del entorno Arduino, los comandos de programación básicos y su utilidad.

e. Aprendizaje de uso de Servomotores

- **Objetivo:** Aprender el funcionamiento de los servos y los comandos necesarios en lenguaje arduino para utilizarlos. También se estudió como conseguir “trucar” un servo, es decir, que un servo que rote 180º tras el trucaje lo haga para 360º.

f. Aprendizaje de uso de Motores DC

- **Objetivo:** Instruirse en los materiales, circuitos y comandos necesarios para conseguir que funcionen motores DC a través del entorno Arduino.

g. Otros

- **Objetivo:** Familiarizarse con los diferentes sistemas de control: redes neuronales, controladores PID, etc.

2. Investigación

Una vez conseguidos unos conocimientos generales de lo necesario, se pasa a observar los robots existentes, estudiar los objetivos y elegir en qué modelos podemos usar de base o tener en cuenta.

- **Duración:** 2 semanas
- **Tiempo empleado:** 6h/semana

3. Diseño

En esta sección se trata de diseñar el vehículo imprimible utilizando las herramientas anteriores. Es una de las etapas más largas, puesto que había un diseño constante para intentar mejorar cada pieza e intentar conseguir la máxima eficiencia del vehículo.

- **Duración:** 4 meses
- **Tiempo empleado:** 30h/semana (incluye las horas de impresión)

Esta sección también podemos dividirla en dos objetivos:

a. Diseño parte delantera

- **Objetivo:** Conseguir diseñar un sistema delantero con suspensión y dirección para el vehículo.

b. Diseño parte trasera

- **Objetivo:** Lograr diseñar un sistema trasero con dos motores DC.

4. Electrónica

Para conseguir mover los motores se requiere hacer un circuito adicional donde debemos incorporar el circuito integrado L293B, que se comuniquen con la placa Arduino.

- **Duración:** 1 semana
- **Tiempo empleado:** 10h/semana

Las partes de las que consta son las siguientes:

a. Diseño circuito

- **Objetivo:** Diseñar el circuito que contiene el elemento L293B y posicionar correctamente todos los elementos requeridos en una placa de puntos.

b. Soldadura

- **Objetivo:** Soldar todos los elementos en la placa de puntos

5. Programación

Ahora debemos comunicar cada una de las partes mecánicas y electrónicas mediante la programación en los diferentes entornos de los que se dispone.

- **Duración:** 1 mes
- **Tiempo empleado:** 10h/semana

La programación se divide a su vez en tres puntos diferentes:

a. Programación C++

- **Objetivo:** Conseguir establecer conexión entre el gamepad y la placa Arduino.

b. Programación Arduino

- **Objetivo:** Procesar los datos recibidos del gamepad y traducirlos a movimientos en el vehículo.

c. Comunicación Bluetooth

- **Objetivo:** Conseguir que la comunicación gamepad arduino se establezca mediante conexión inalámbrica Bluetooth.

6. Pruebas

La sección más importante donde se comprueba la correcta implementación de las tareas anteriores.

- **Duración:** 2 mes
- **Tiempo empleado:** 20h/semana

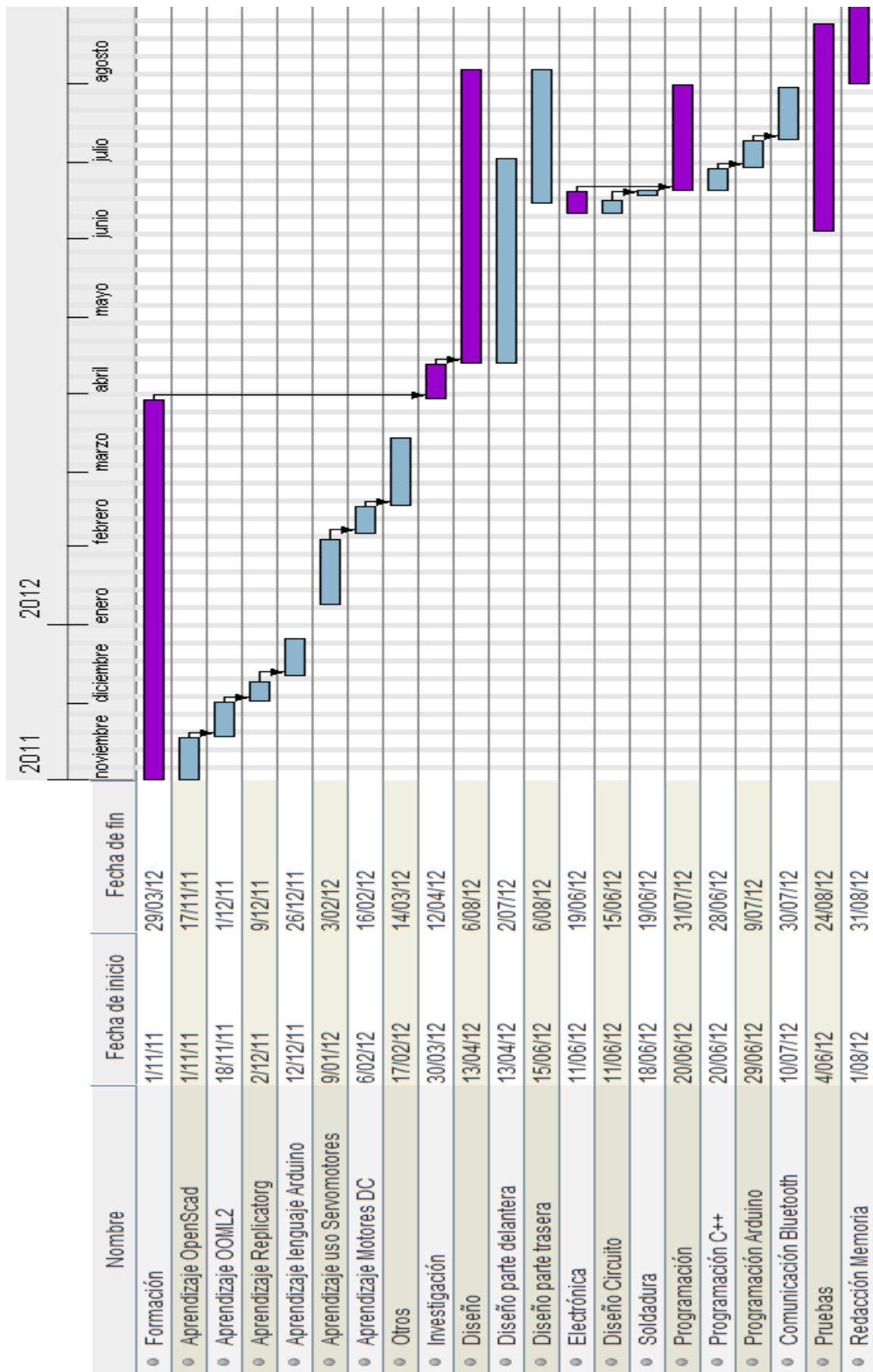
7. Redacción de la memoria

Como última tarea, la redacción de todos los pasos, tareas y materiales utilizados en el proyecto de forma detenida y detallada.

- **Duración:** 1 mes
- **Tiempo empleado:** 25h/semana

Todo ello puede verse resumido en la siguiente tabla y en el gráfico de Gantt (página siguiente):

Tarea	Duración (semanas)	Tiempo empleado (h/semana)	Total (horas)
1. Formación	20	2	40
2. Investigación	2	6	12
3. Diseño	16	30	480
4. Electrónica	1	10	10
5. Programación	4	10	40
6. Pruebas	8	20	160
7. Redacción memoria	4	25	100
Total			842



B.2 Presupuesto

En este apartado se especificarán los costes tanto del material requerido para el proyecto, tanto software como hardware.

<i>Medios empleados</i>	<i>Unidades</i>	<i>Precio Unitario (€)</i>	<i>Total</i>
Software			
Sistema Operativo Ubuntu	1	0	0
Compilador Qt Creator	1	0	0
OpenScad	1	0	0
Replicatorg	1	0	0
Arduino 1.0.1	1	0	0
Total			0
Hardware			
Rodamientos	2	1	2
Placa Arduino Leonardo	1	23,75	23,75
Módulo Bluetooth	1	8,25	8,25
Batería LIPO 2S 1000mAh 20C	1	12	12
Placa metacrilato 40 x 30 cm	1	12,45	12,45
Circuito L293B	1	4,5	4,5
Placa de Puntos	1	1,5	1,5
Motores DC	2	0	0
Servomotor	1	4	4
Gamepad Xbox 360 USB	1	30	30
Total			98,45
Total			98,45

El coste total del vehículo serían **98,45€**.

Se aprecia que no es muy barato, pero si hiciéramos un segundo vehículo, con los materiales de los que ya disponemos, nos ahorraríamos la placa de metacrilato (puesto que con las que venden normalmente se podrían hacer al menos 3 bases). Se podría adquirir un gamepad sin marca con las mismas características que el anterior por unos 20€, y podríamos usar la placa SkyMega (coste 12 €) en lugar de Arduino, de forma que el coche podría costarnos 66 €.

